Systemy wieloagentowe (Multi Agent Systems - MAS)

– aspekty wybrane –

Politechnika Gdańska
Wydział Elektrotechniki i Automatyki

Opracowanie: dr inż. Tomasz Rutkowski
Katedra Inżynierii Systemów Sterowania
2014/2015

Plan prezentacji

- Czym są systemy wieloagentowe?
- Agent, środowisko, aspekty komunikacji agentów

- Typowe struktury sterowania a systemy wieloagentowe

- Przykłady zastosowania systemów wielooagentowych:
  - Systemy sterowania:
    - Wielooagentowy system sterowania komfortem termicznym w budynku jednorodzinnym
  - Obliczenia, harmonogramowanie:
    - Algorytmy genetyczne/ewolucyjne
    - Algorytmy rojowe (rój cząstek, kolonia mrówek)

Czym są systemy wieloagentowe?

Czym są systemy wieloagentowe?

- Systemy wieloagentowe są to systemy złożone z komunikujących się i współpracujących ze sobą agentów, realizujących wspólne cele

- Systemy wieloagentowe znajdują zastosowanie w rozwiązywaniu problemów o charakterze rozproszonym lub złożonym obliczeniowo

Czym są systemy wieloagentowe?

- Przykładami zastosowania systemów wieloagentowych są:
  - wyszukiwanie informacji w sieci internetowej,
  - zarządzanie sieciami telekomunikacyjnymi,
  - kontrola ruchu lotniczego,
  - rozwiązywanie zadań NP-trudnych,
  - …
- Charakterystyczną cechą systemów wieloagentowych jest ich różnorodność wynikająca z różnorodności dziedzin obliczeń, do których są stosowane

Czym jest agent?

- Nie ma jednej precyzyjnej definicji agenta (oto dwie przykładowe)

Definicja 1

Agentem jest element służący do budowy systemów zdecentralizowanych. Agent jest autonomiczny i działa (posiada inicjatywę) w oparciu o obserwację docierające do niego z otoczenia (środowiska).

Definicja 2

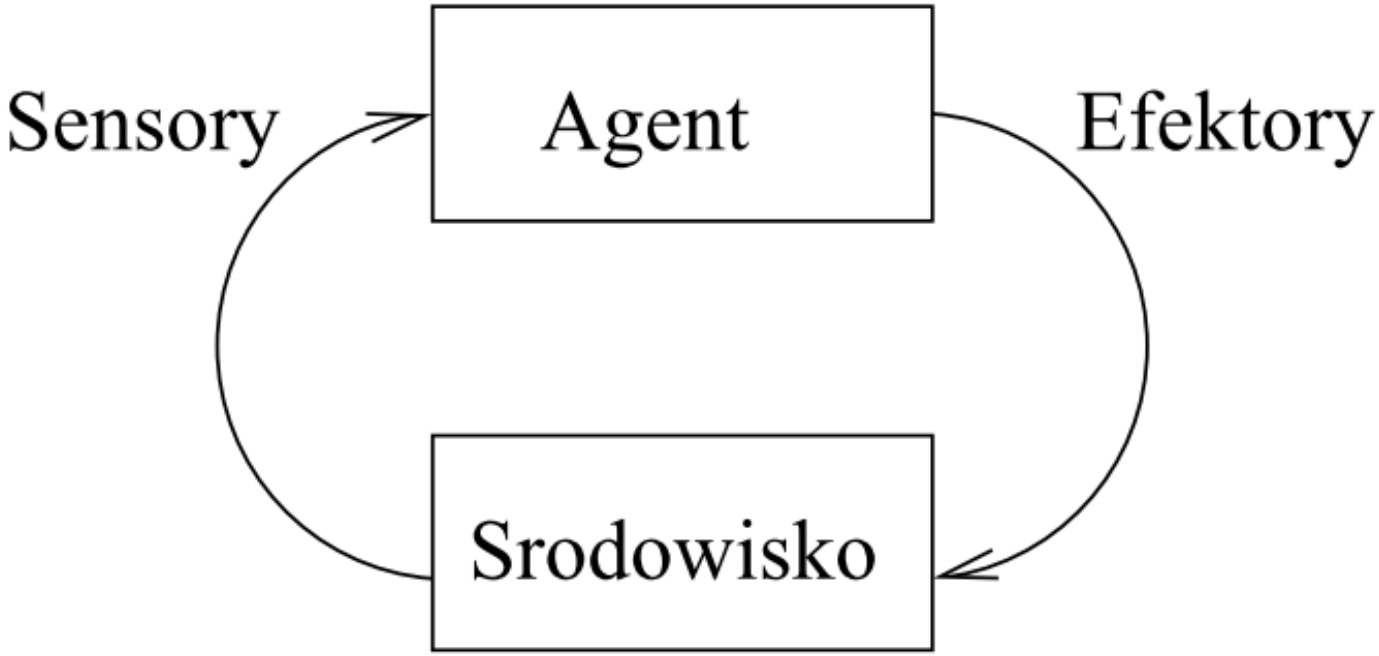
Agentem jest cokolwiek, co postrzega środowisko i w jakimś sposób na nie odziaływuje.

Czym jest agent?

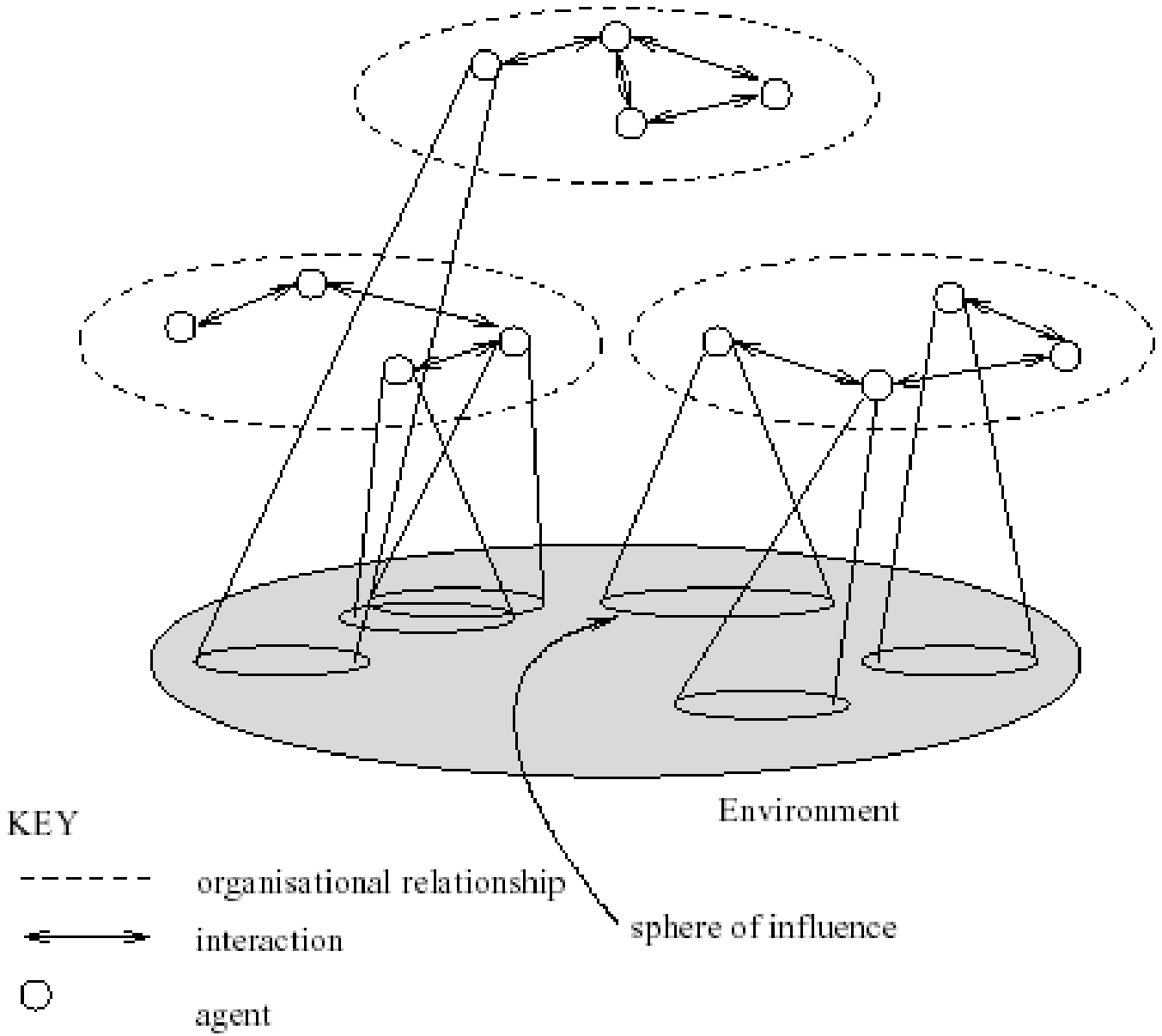
- Typowo za agenta uznaje się systemem komputerowy (program), usytuowany w określonym środowisku, mogący korzystać z określonych zasobów, zdolny do autonomicznego działania, w celu osiągnięcia określonych celów i posiadający motywację do działania

Czym jest agent?

- Agent a środowisko



Czym jest agent?



Czym jest środowisko?

- Jak opisać środowisko pracy agenta:
  - dostępne / niedostępne: określa dostępność informacji o środowisku, im bardziej „dostępne” tym „prościej” zbudować agenta,
  - deterministyczne /niedeterministyczne: deterministyczne środowisko oznacza, że każda akcja agenta ma jednoznacznie określoną, gwarantowaną odpowiedź środowiska,
  - epizodyczne /nieepizodyczne: w środowisku epizodycznym zmiany środowiska nie zależą od działania innych agentów,
  - statyczne /dynamiczne: w środowisku statycznym jest skończona liczba możliwych akcji i percepcji agenta.

Czym jest agent?

- Cechy agenta:
  - autonomiczność: agenci są w stanie działać samodzielnie
  - reaktywność: agenci są w stanie zidentyfikować środowisko i zareagować w sposób umożliwiający realizację zadania,
  - pro-aktywność: agenci są w stanie przejąć inicjatywę w celu realizacji zadania,
  - zdolność współpracy: agenci są w stanie współdziałać ze sobą w celu realizacji zadania.

Czym jest agent?

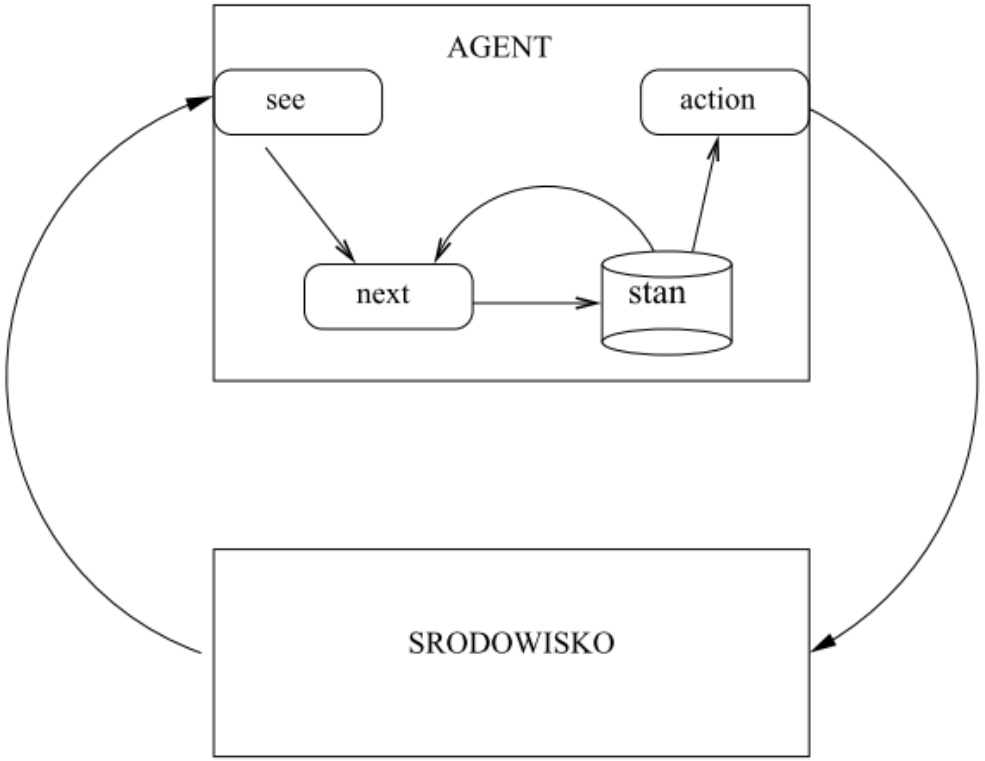
- Typowo rolę agenta w systemie wielooagentowym porównuje się do roli obiektów w programowaniu obiektowym, niemniej jednak są między nimi różnice:
  - agent posiada wewnętrzną świadomość i swój własny cel, który może być różny od celu innego agenta, w związku z tym prośba od innego agenta zostanie wykonana tylko wtedy jeżeli jest ona zgodna z interesem pytanego agenta *(obiekty mają metody publiczne, które umożliwiają innym obiektom uruchamianie ich niezależnie od woli obiektu posiadającego taką metodę)*,
  - agenci mają możliwość dostosowania swojego zachowania do sytuacji: reaktywność, proaktywność, zdolność współpracy *(obiekt też może mieć takie właściwości, ale są one niezmienne i nie zależą od stanu innych obiektów)*,
  - każdy agent ma co najmniej jeden wewnętrzny wątek sterujący jego stanem wewnętrznym.

Percepcja agenta

$$see: \; S \rightarrow P$$

$$action: \; P^* \rightarrow A$$

Percepcja agenta



$$see: \; S \rightarrow P$$

$$next: \; I \times P \rightarrow I$$

$$action: \; I \rightarrow A$$

Percepcja agenta

- Podstawowe typy agentów w zależności od sposobu realizacji funkcji decyzyjnych (*action*) agenta:
  - agent logiczny: funkcja decyzyjna implementowana jest w wyniku dedukcji
  - agent reaktywny: funkcja decyzyjna jest implemenmtowana na podstawie analizy stanu środowiska (funkcje *IF-THEN*)
  - agent BDI (ang. beliefs, desires, intentions): funkcja decyzyjna zależy od kombinacji danych reprezentowanych jako *przekonanie-pragnienie-intencja*
  - agenci o architekturze warstwowej: funkcja decyzyjna jest realizowana przez różne warstwy programowe, z których każda odpowiedzialna jest za inny poziom abstrakcji środowiska

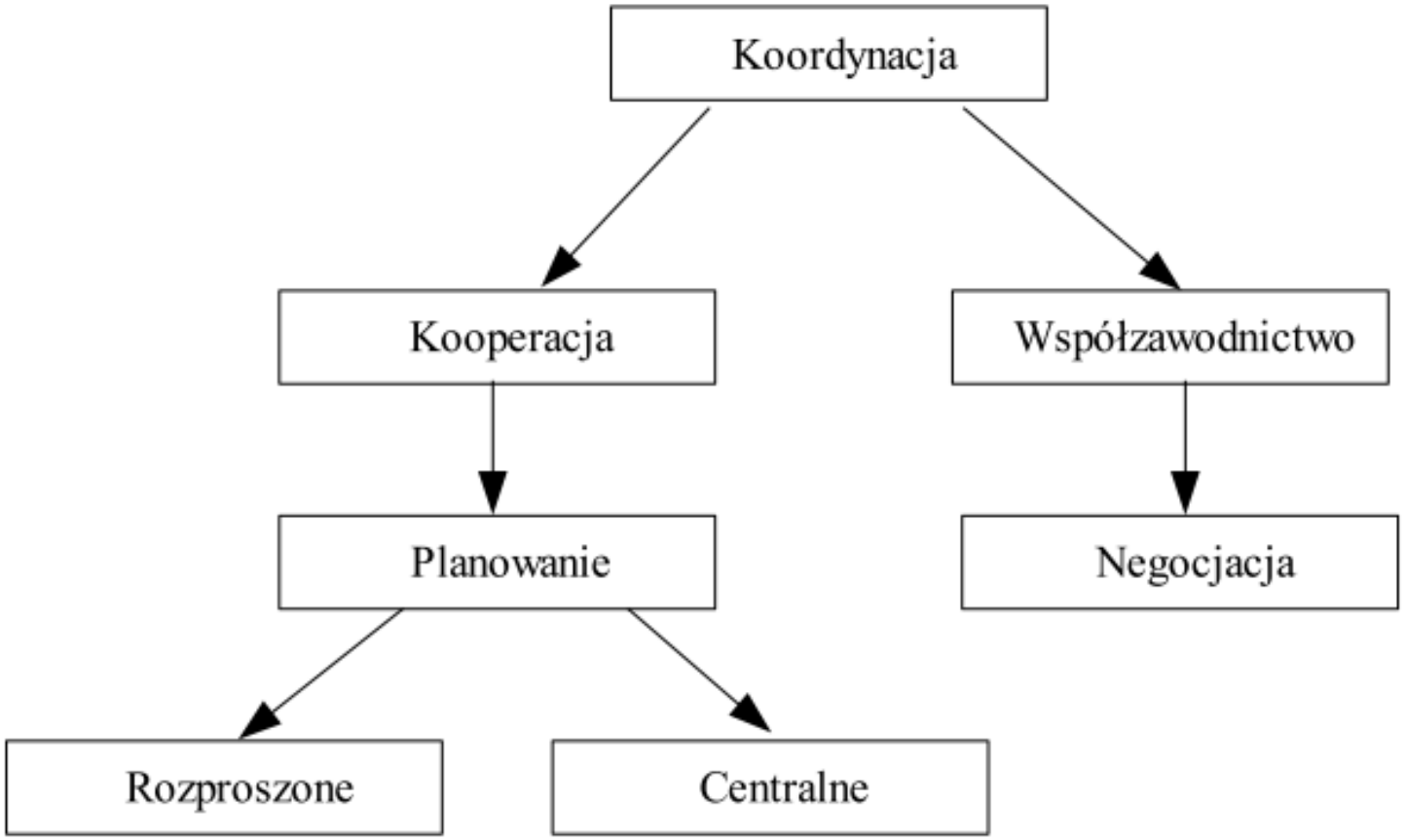
System wieloagentowy

- System wieloagentowy posiada następujące elementy:
  - środowisko,
  - obiekty umieszczone w środowisku,
  - agenci operujący w środowisku,
  - relacje pomiędzy agentami.

- System wieloagentowy posiada następujące cechy:
  - składa się z autonomicznych agentów, z których każdy realizuje swoje cele, które mogą lecz nie muszą być wspólne z interesem grupy,
  - jest otwarty i podejmowanie decyzji jest rozproszone (agenci mogą posiadać zróżnicowane poziomy autonomii),
  - środowisko systemu wielooagentowego określa rodzaj komunikacji.

System wieloagentowy

- Komunikacja pomiędzy agentami



System wieloagentowy

- Podział agentów ze względu na możliwości komunikacyjne

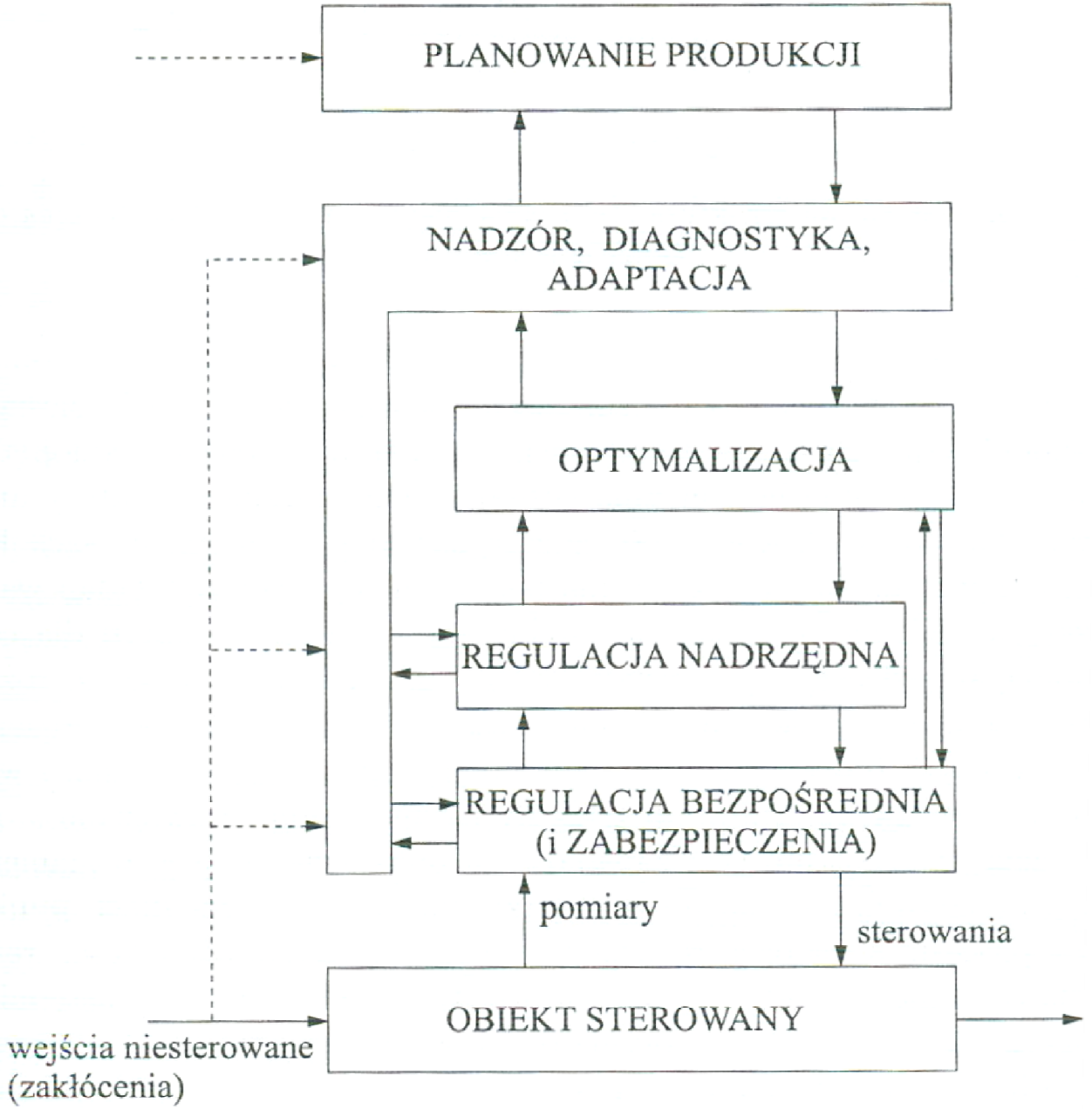
	Agent podstawowy	Agent pasywny	Agent aktywny	Agent równy
Otrzymywanie rozkazów	✔	✔	✔	✔
Otrzymywanie zapytań		✔		✔
Wysyłanie rozkazów		✔	✔	✔
Wysyłanie zapytań			✔	✔

System wieloagentowy

- Charakter wymienianych komunikatów pomiędzy agentami:
  - propozycja określonej akcji,
  - akceptacja akcji,
  - odrzucenie akcji
  - odwołanie akcji,
  - niezgodzenie się na zaproponowaną akcję,
  - kontrpropozycja na zaproponowaną akcję.

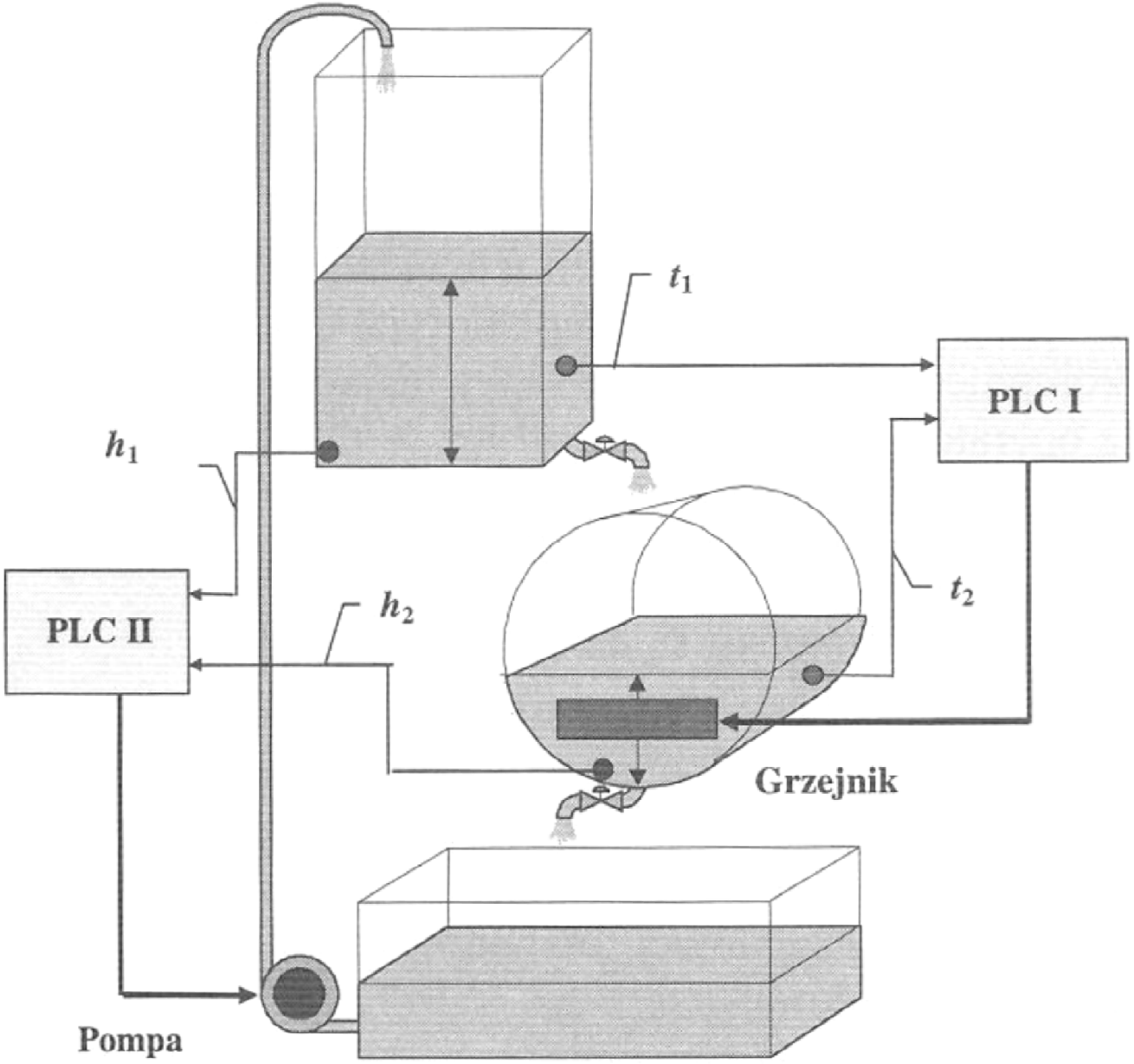
Typowe struktury sterowania a systemy wieloagentowe

Podstawowa warstwowa struktura sterowania



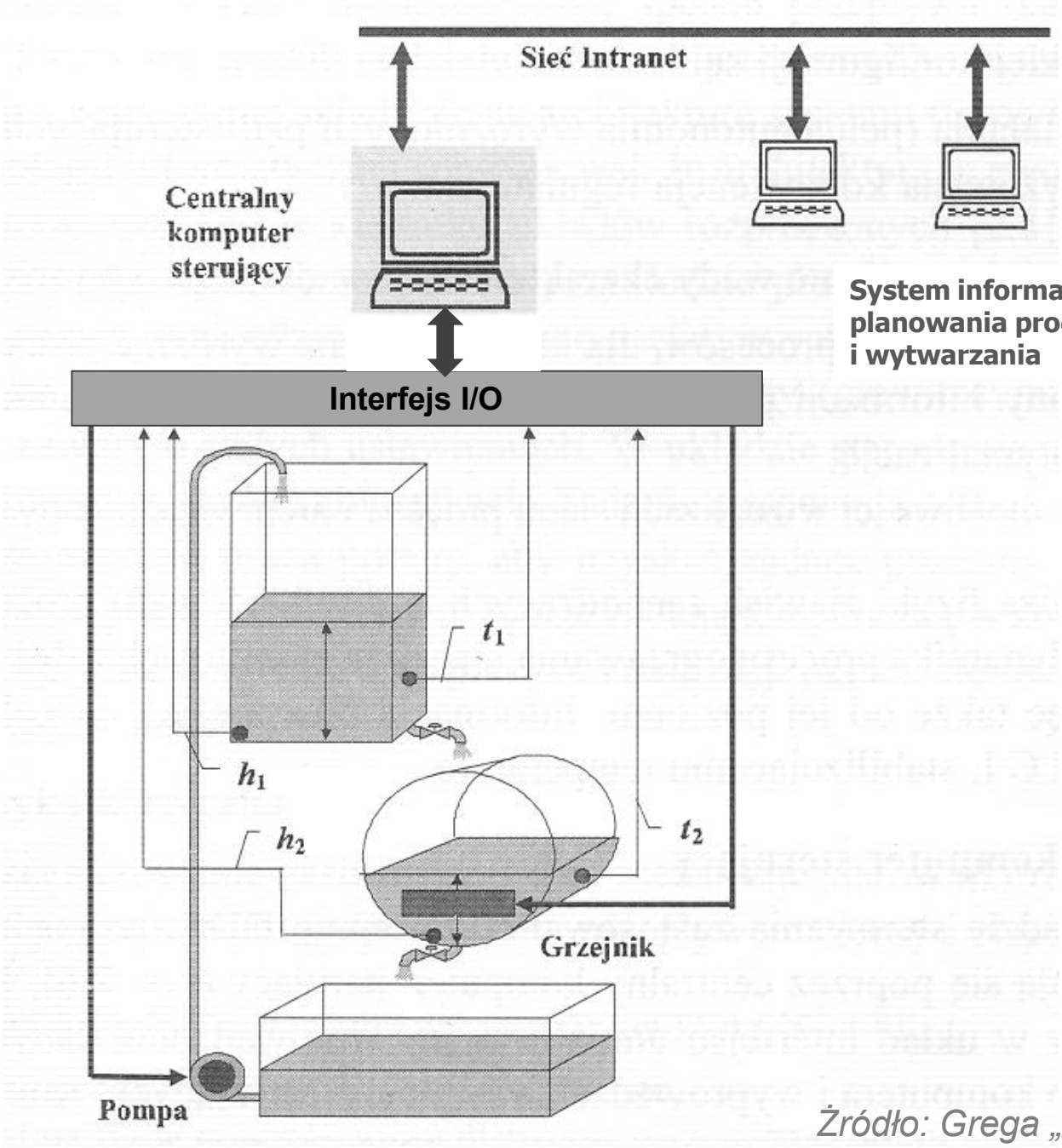
Źródło: P.Tatiewski „Sterowanie...

Przykład klasycznej struktury sterowania



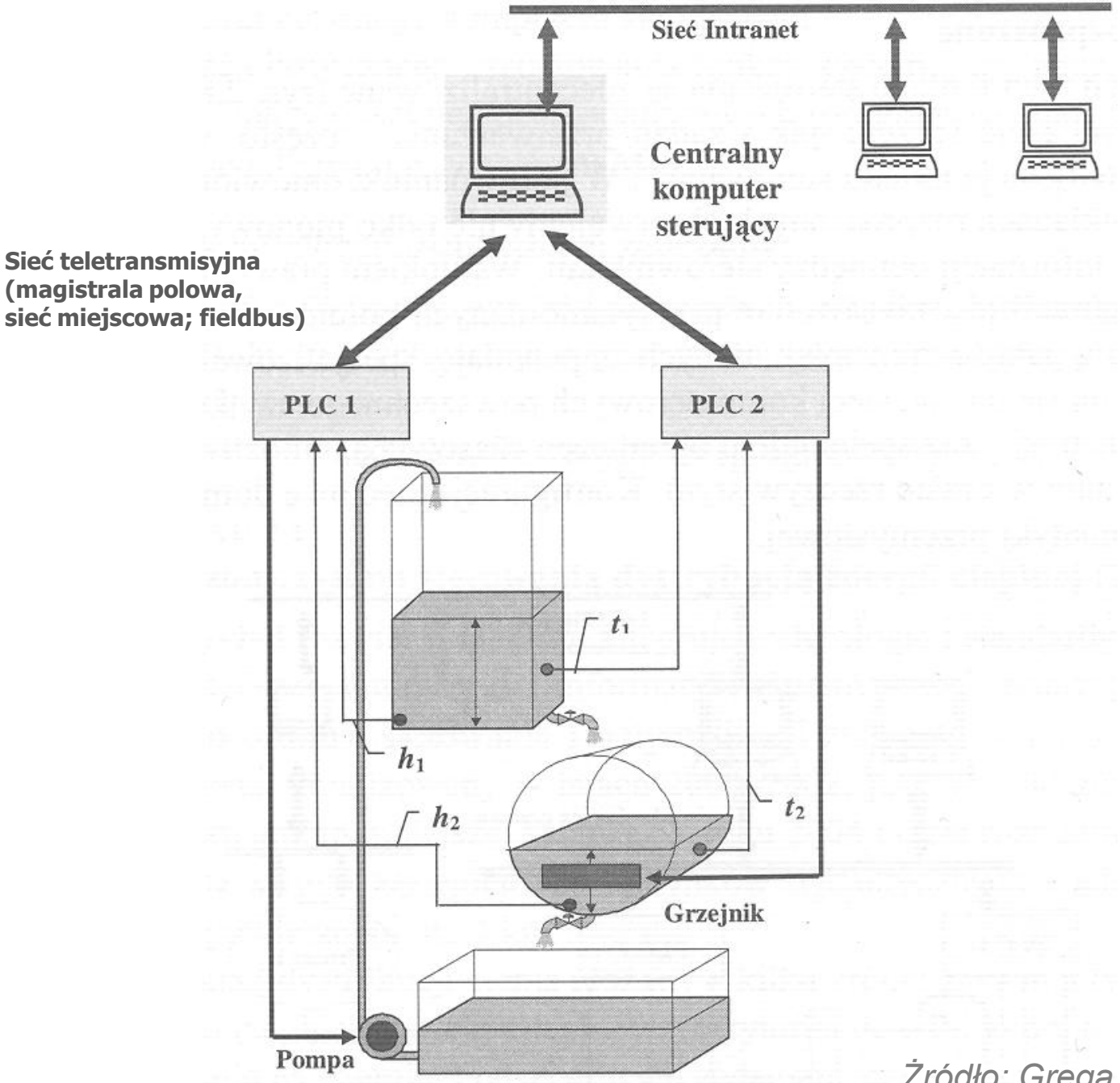
Źródło: Grega „Metody i algorytm

Przykład scentralizowanej struktury sterowania



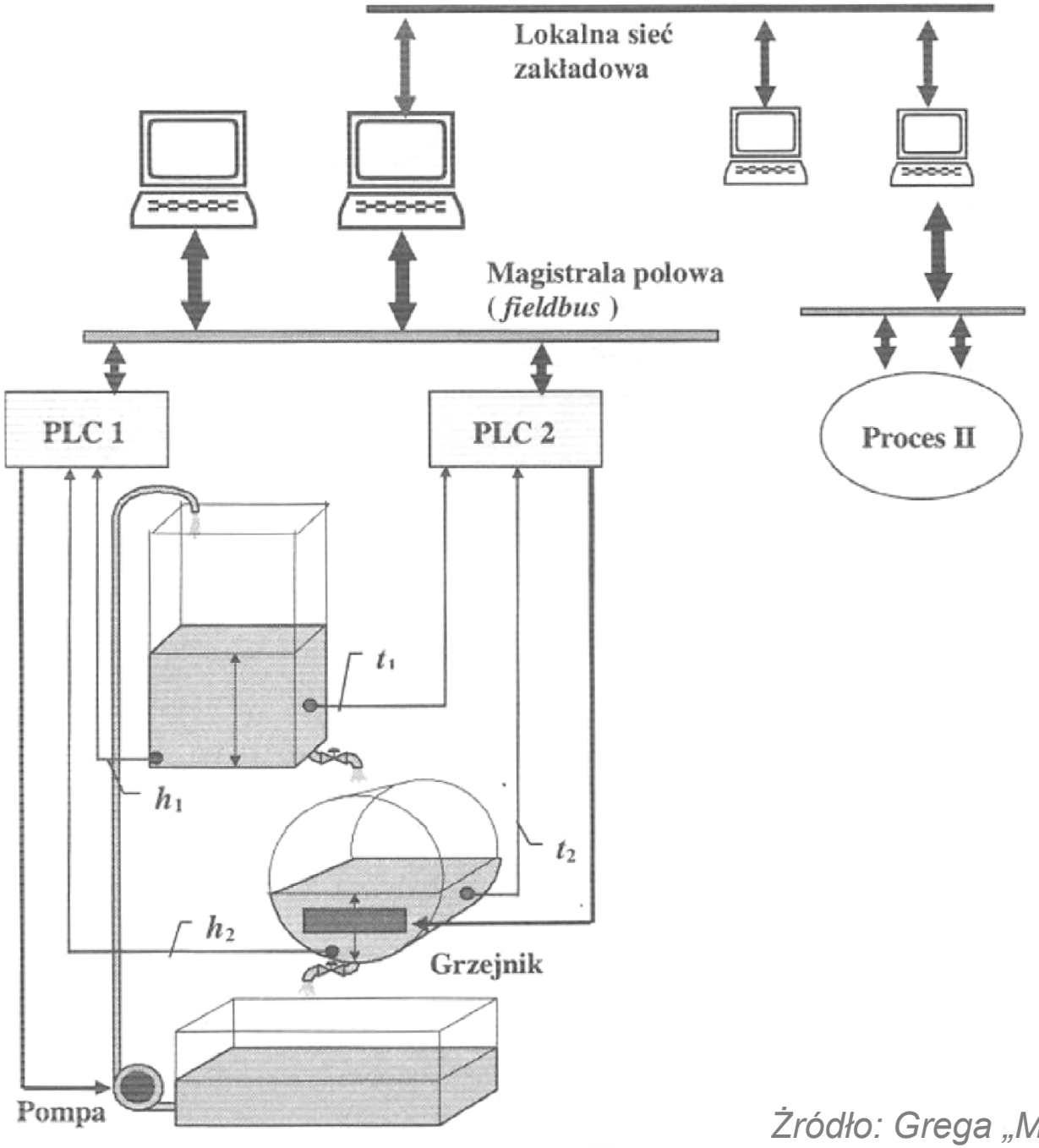
Źródło: Grega „Metody i algorytm

Przykład wielowarstwowej struktury sterowania

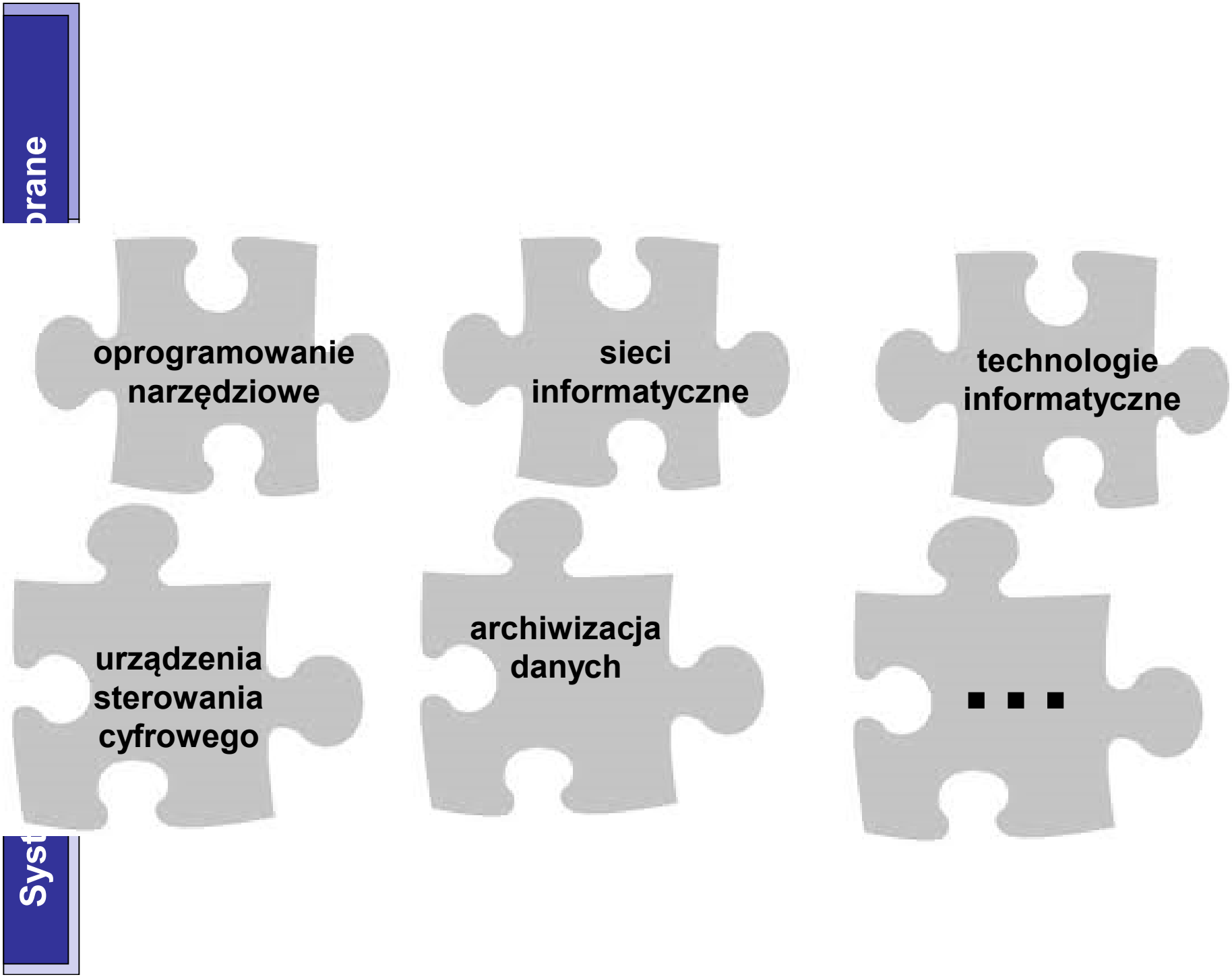


Źródło: Grega „Metody i algorytm

Przykład rozproszonej struktury sterowania



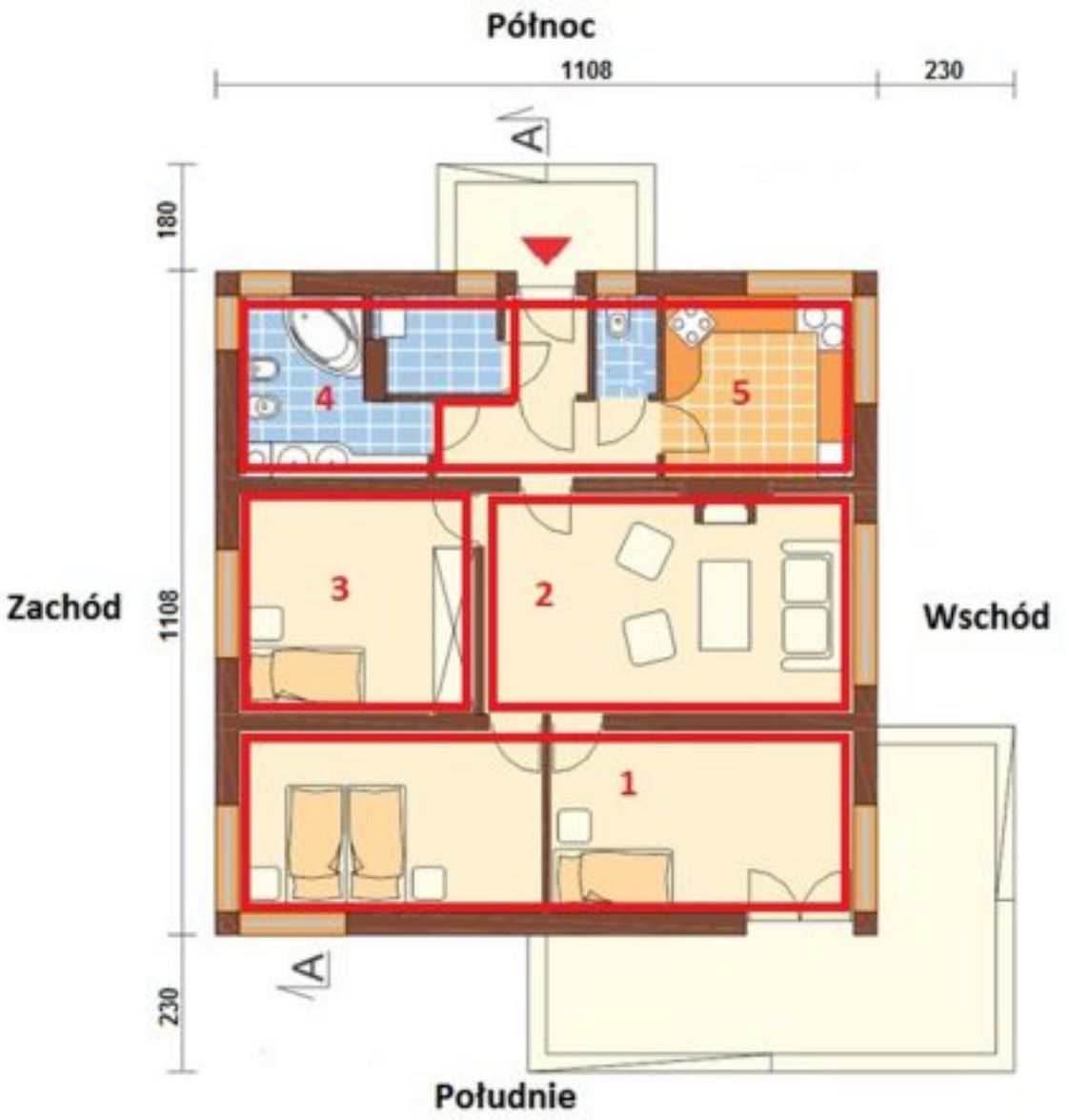
Źródło: Grega „Metody i algorytm


oprane
oprogramowanie narzędziowe
sieci informatyczne
technologie informatyczne
urządzenia sterowania cyfrowego
archiwizacja danych
. . .
Syst

oprogramowanie narzędziowe
technologie informatyczne
DCS
(ang. Distributed Control Systems)
urządzenia sterowania cyfrowego
archiwizacja danych
. . .

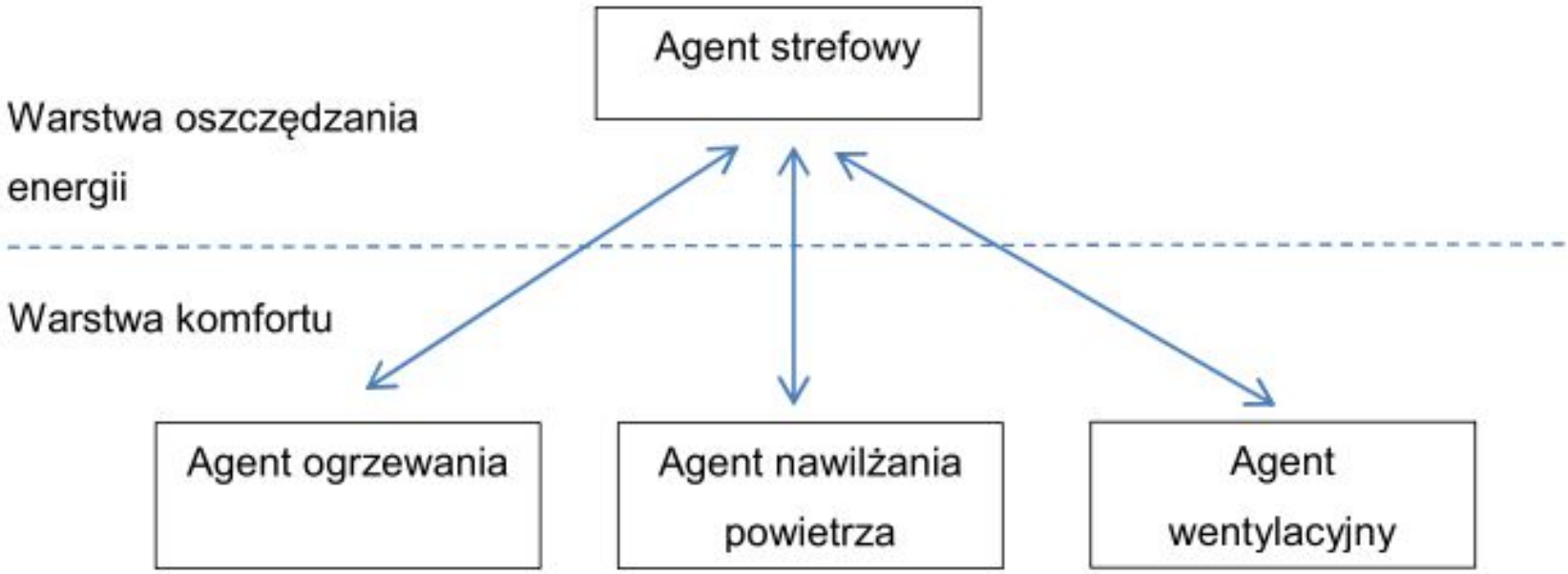
Wielooagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym



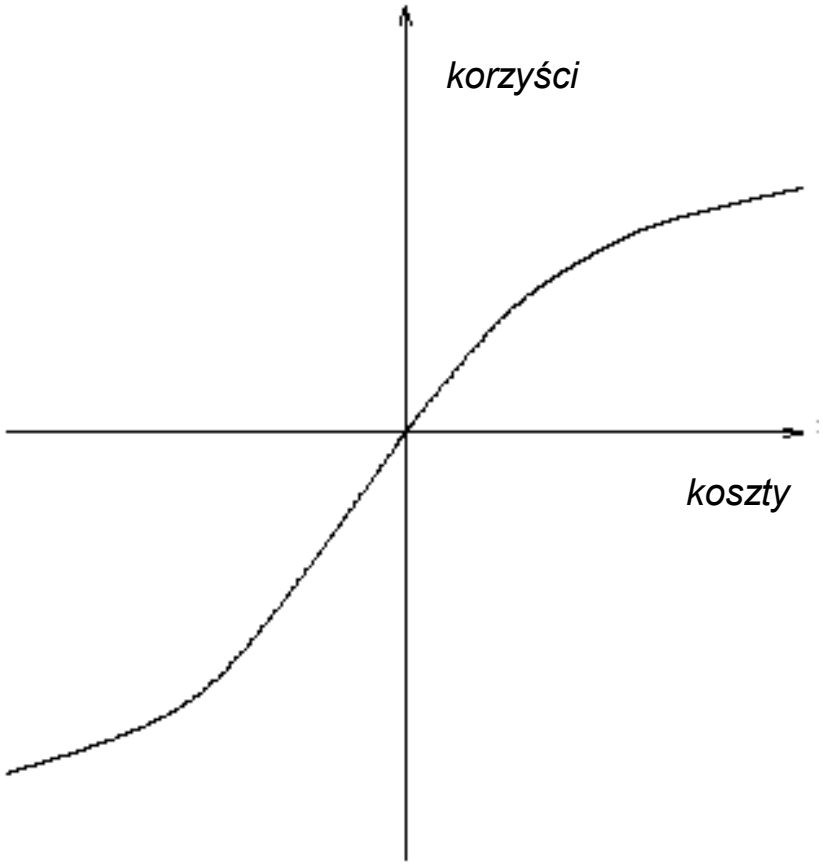
Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

- Agent strefowy podejmuje decyzje na podstawie funkcji użyteczności (innej dla każdego z agentów warstwy komfortu) w której wyodrębniane są dwa składniki:
  - Korzyści,
  - Koszty.

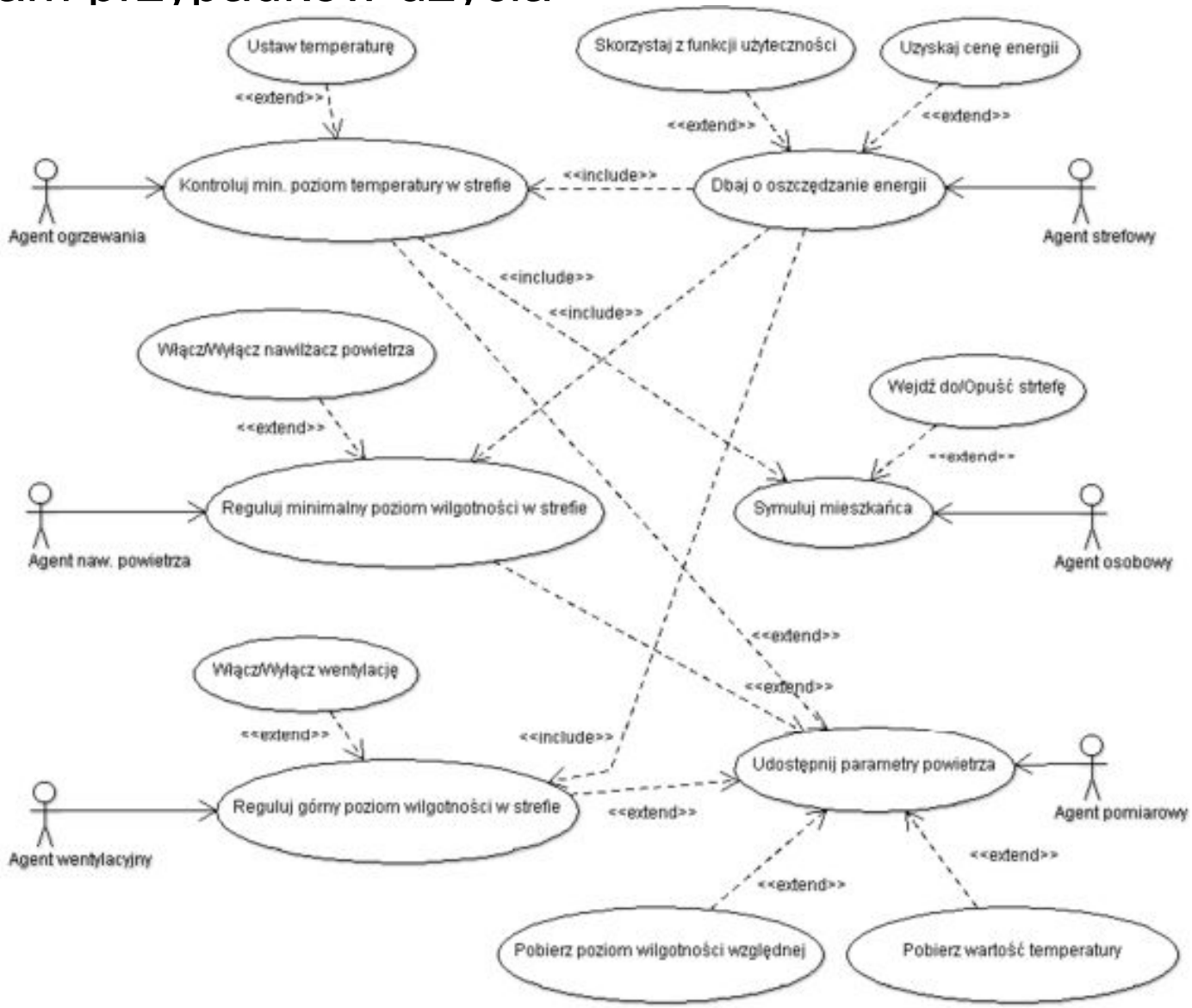


Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

- Zaprojektowany system wielooagentowy opisano za pomocą diagramów:
  - przypadków użycia: do modelowania zadań wszystkich elementów (aktorów) systemów,
  - czynności: do modelowania funkcjonalności poszczególnych elementów systemu,
  - sekwencji: do opisu interakcji pomiędzy elementami systemu wraz z upływem czasu,
  - klas: do opisu statycznych cech modelu.

Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

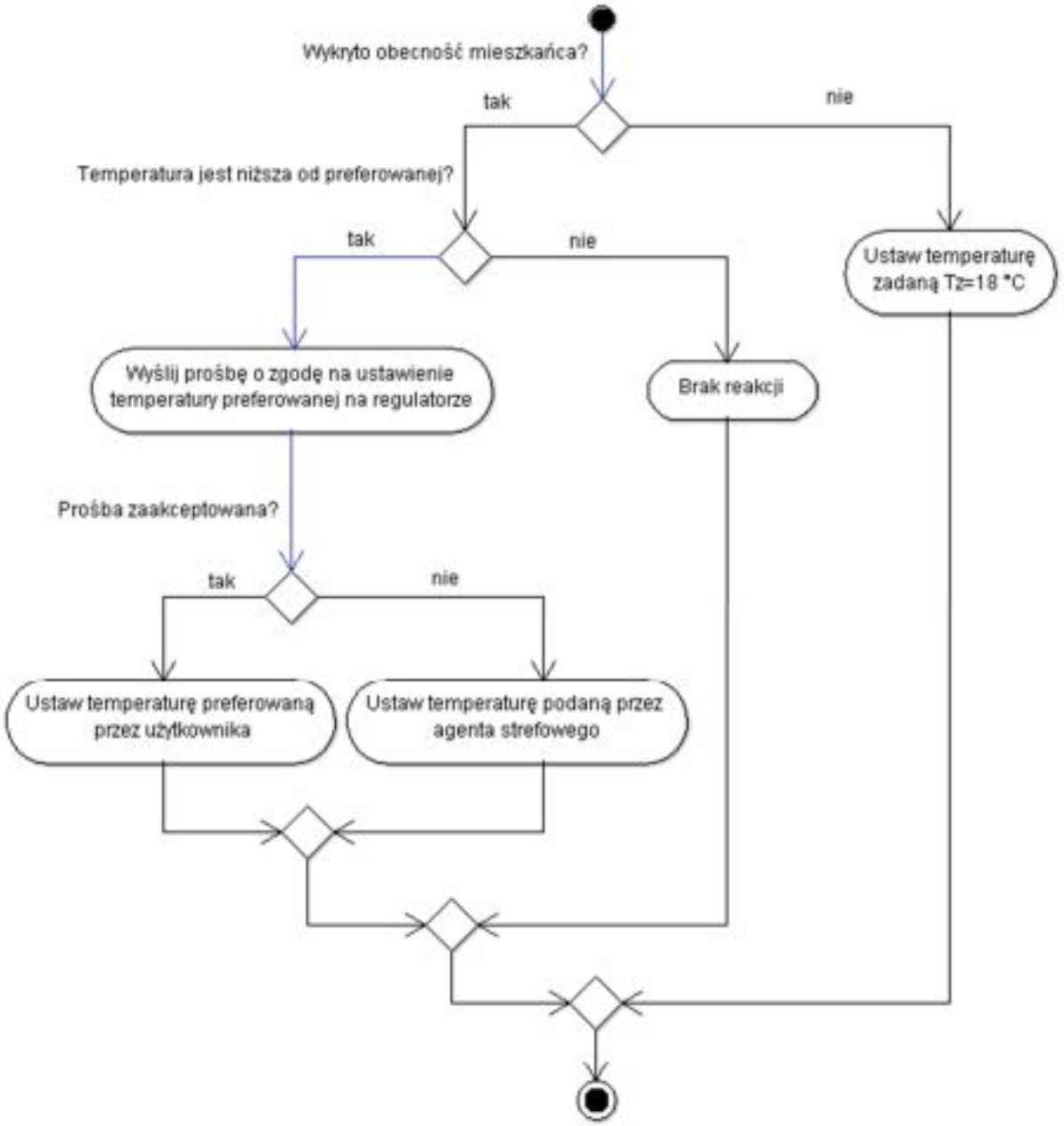
Diagram przypadków użycia



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Diagram czynności:

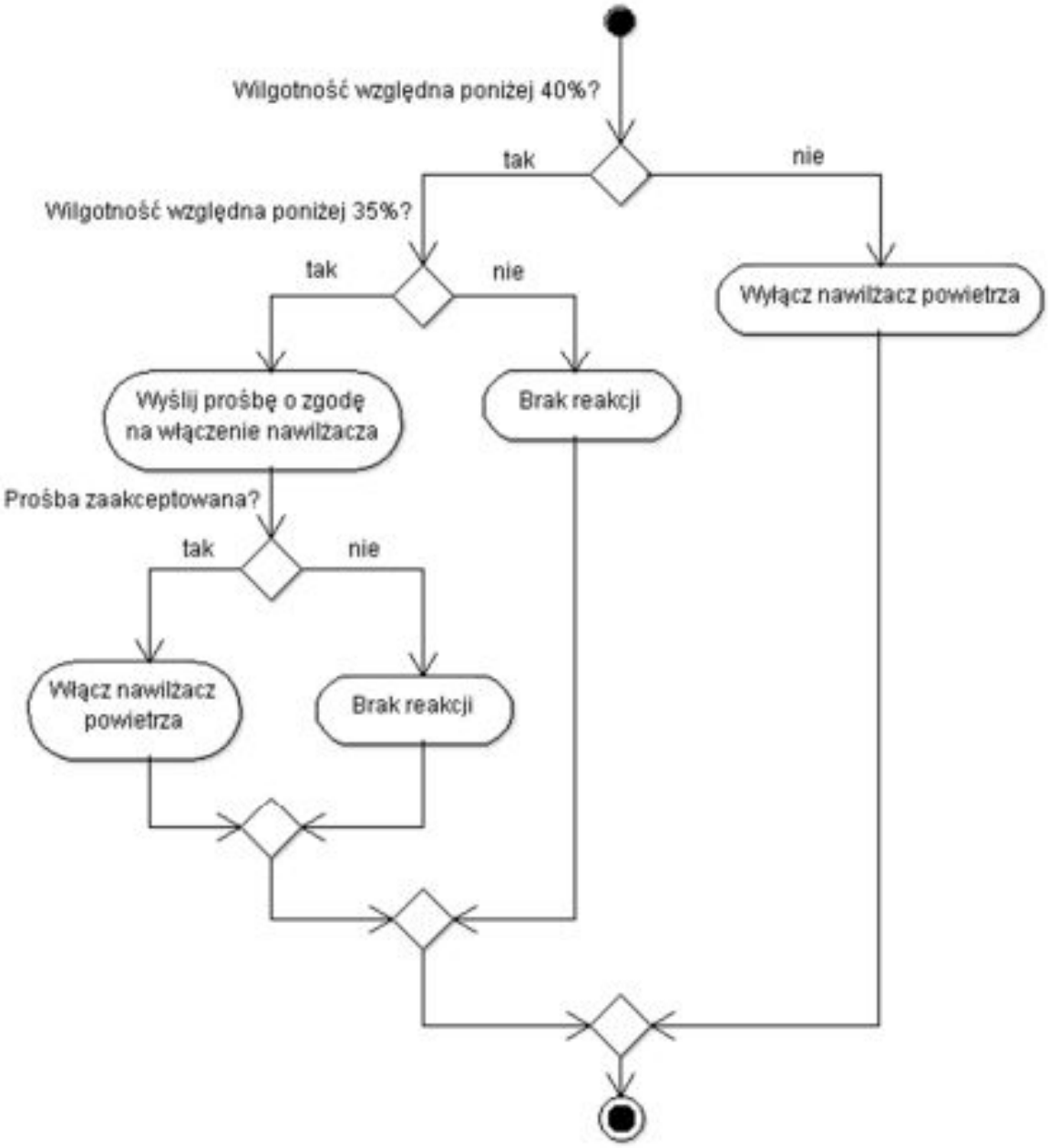
Agent ogrzewania



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Diagram czynności:

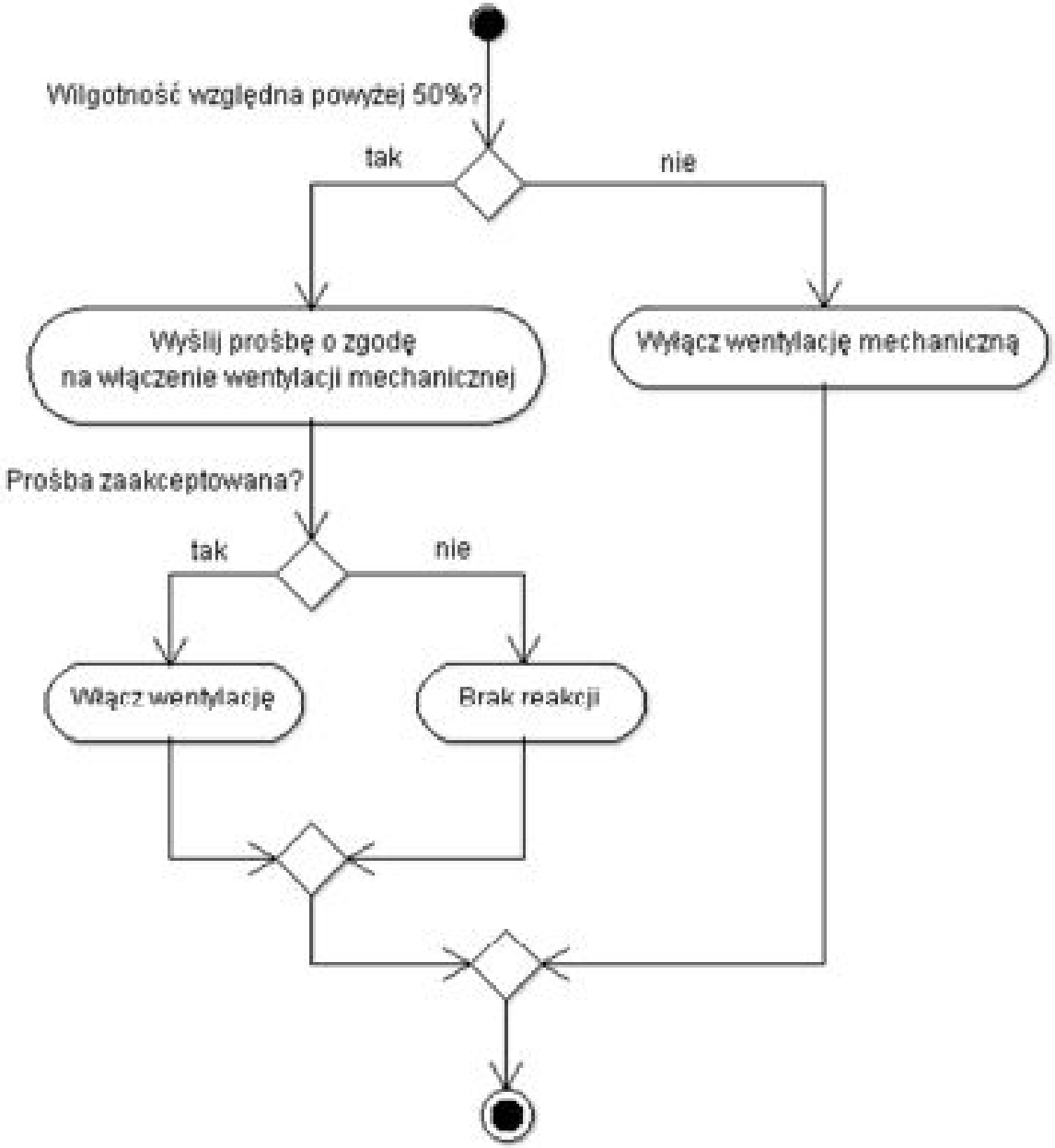
Agent nawilżania powietrza



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Diagram czynności:

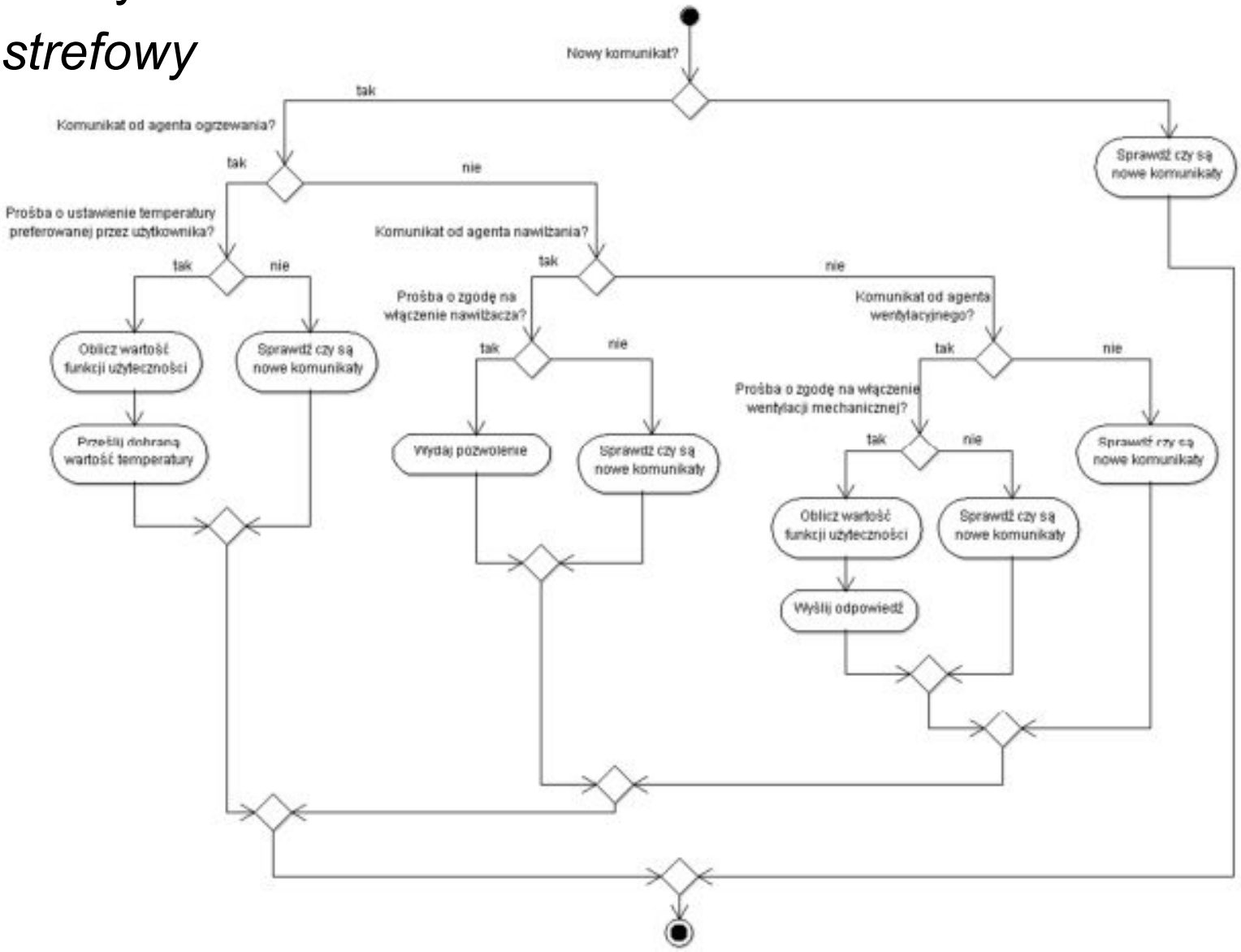
Agent wentylacji



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Diagram czynności:

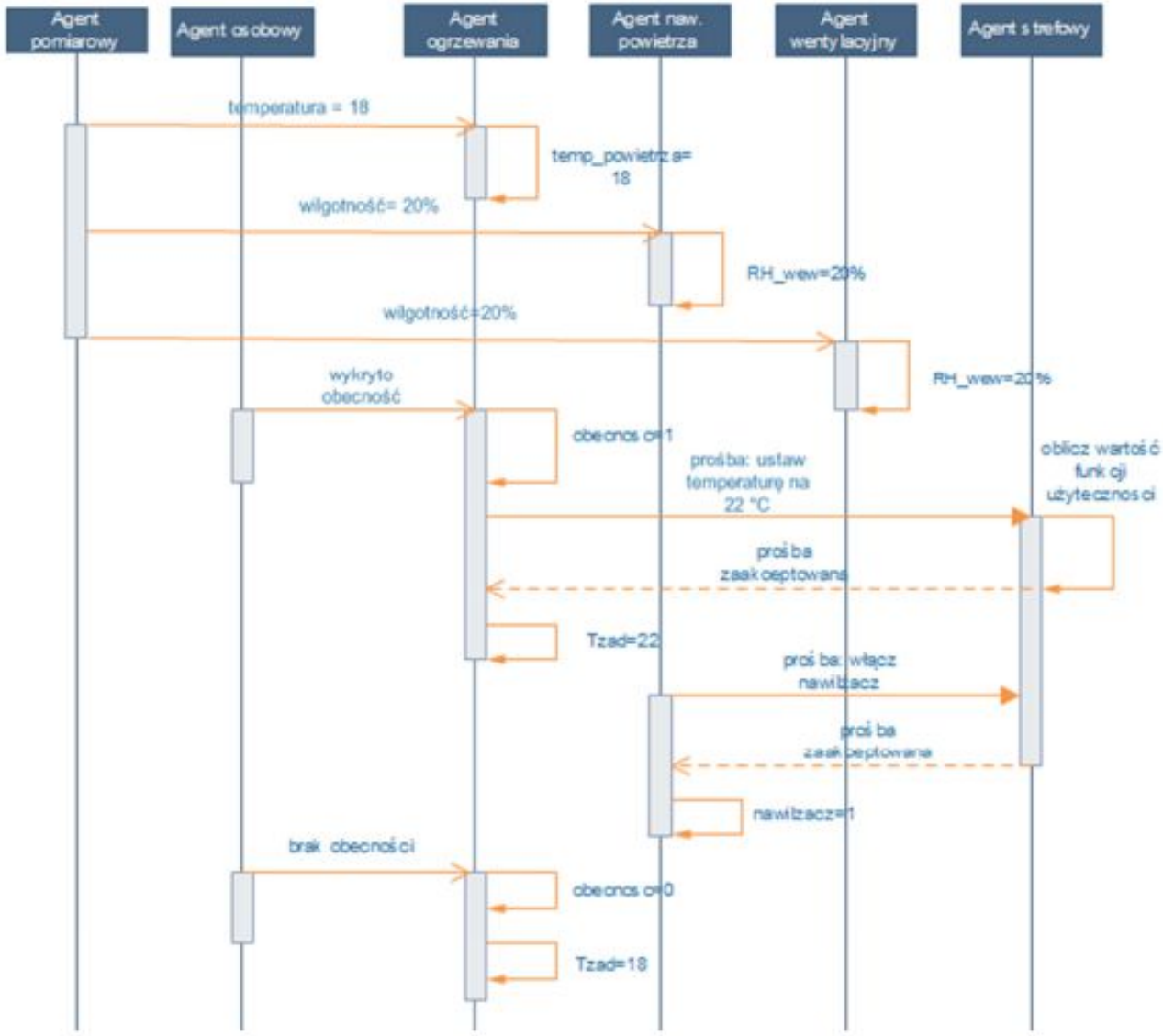
Agent strefowy



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Diagram sekwencji

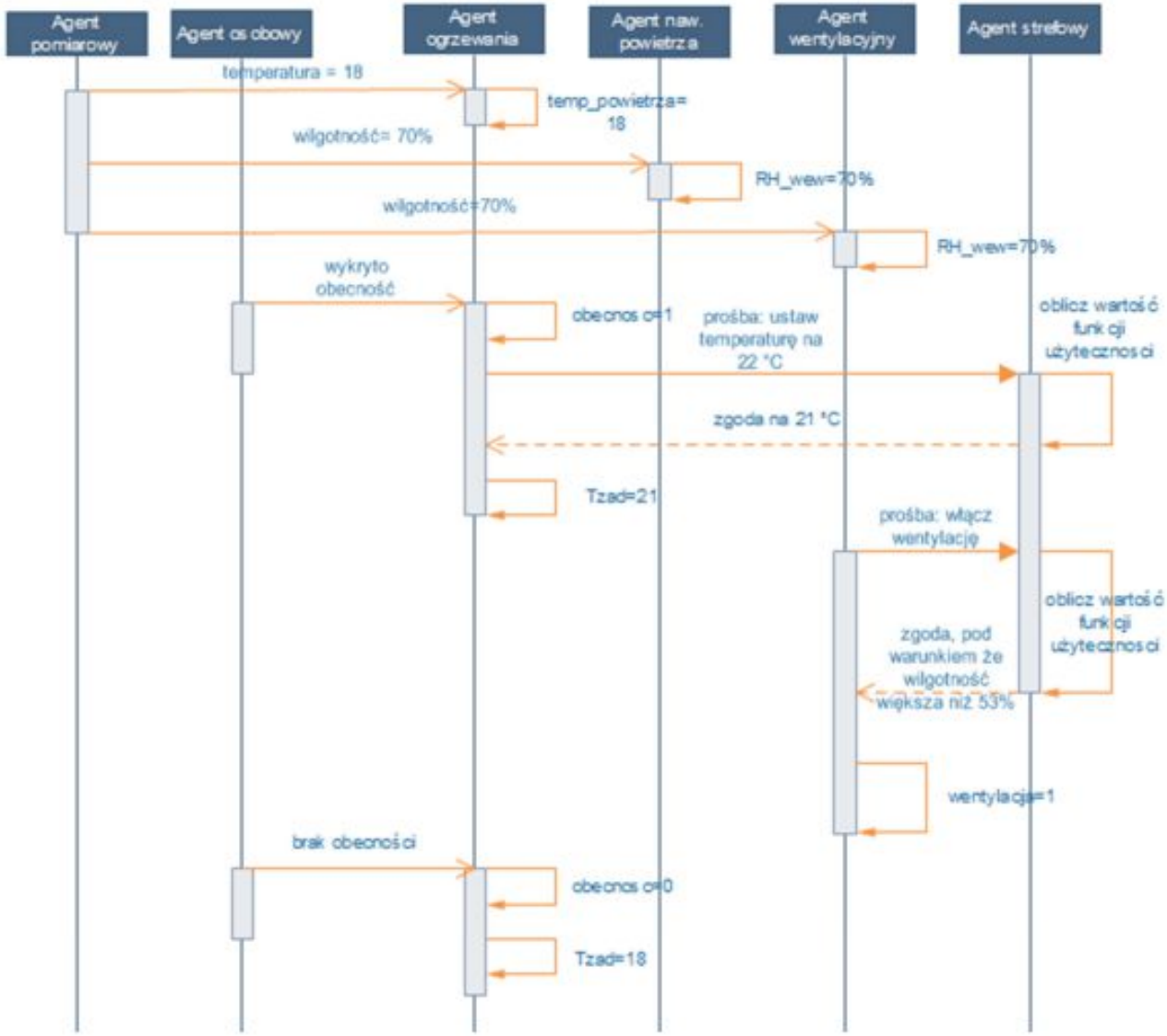
Przykład 1



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

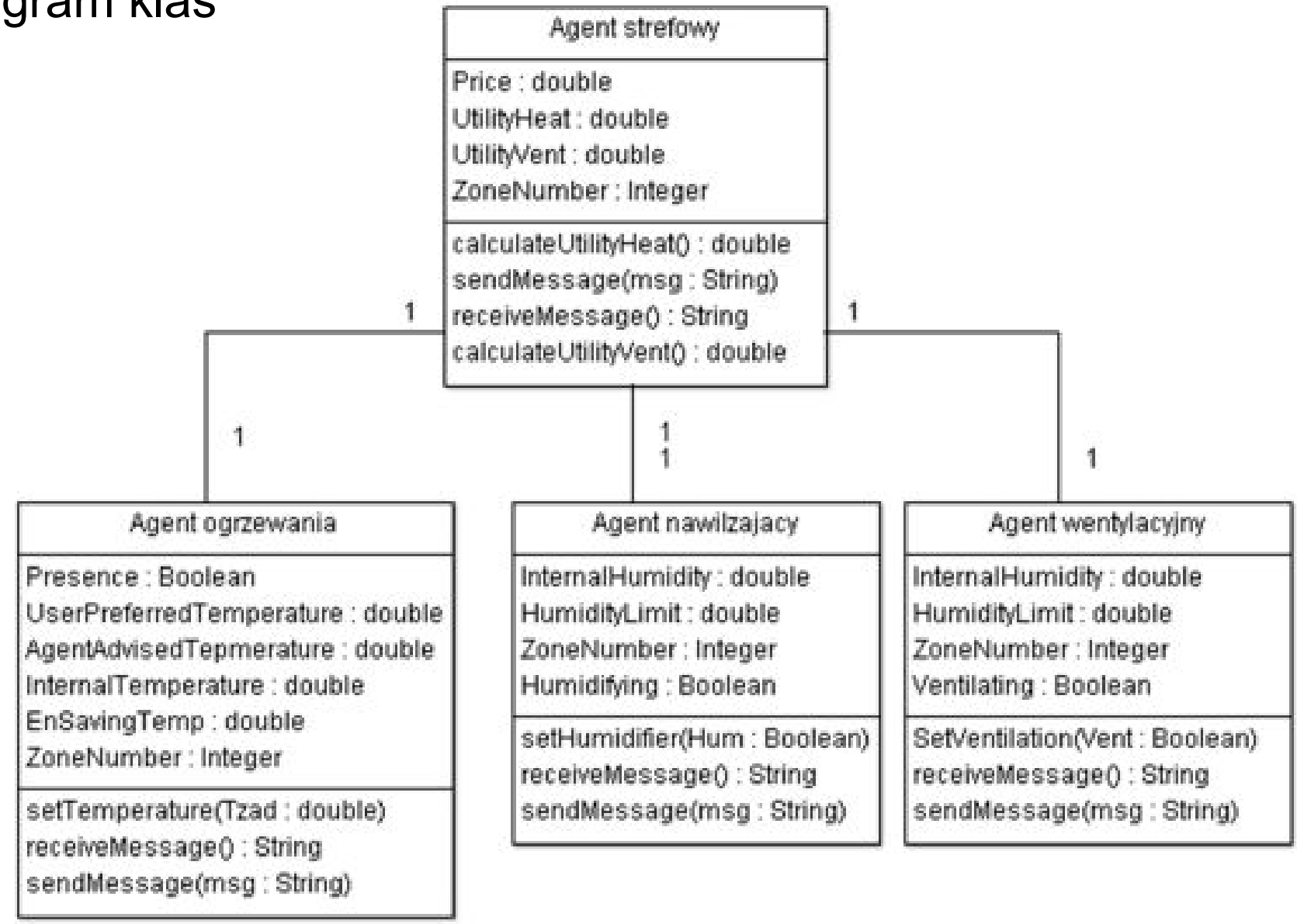
Diagram sekwencji

Przykład 2



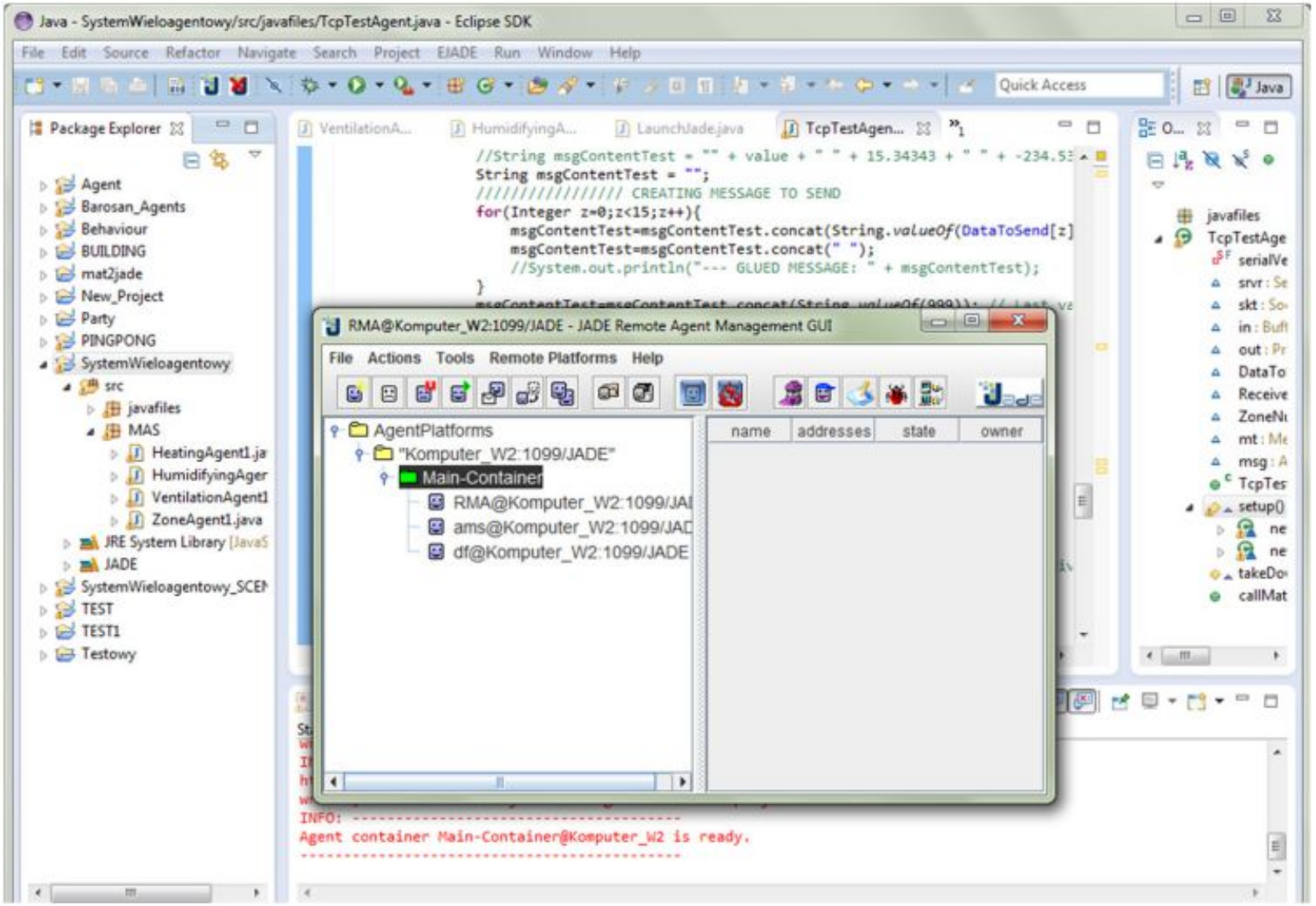
Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Diagram klas

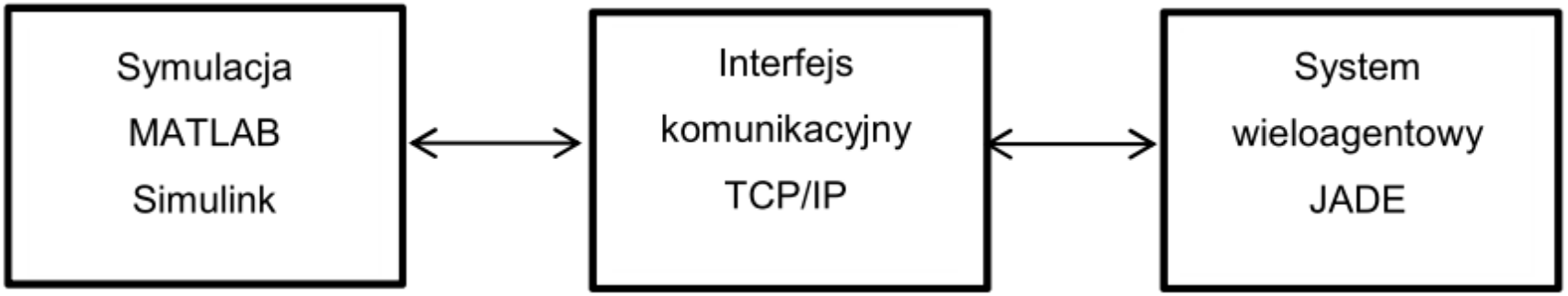


Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

Platforma JADE (JAVA Agent DEvelopment Framework)



Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym



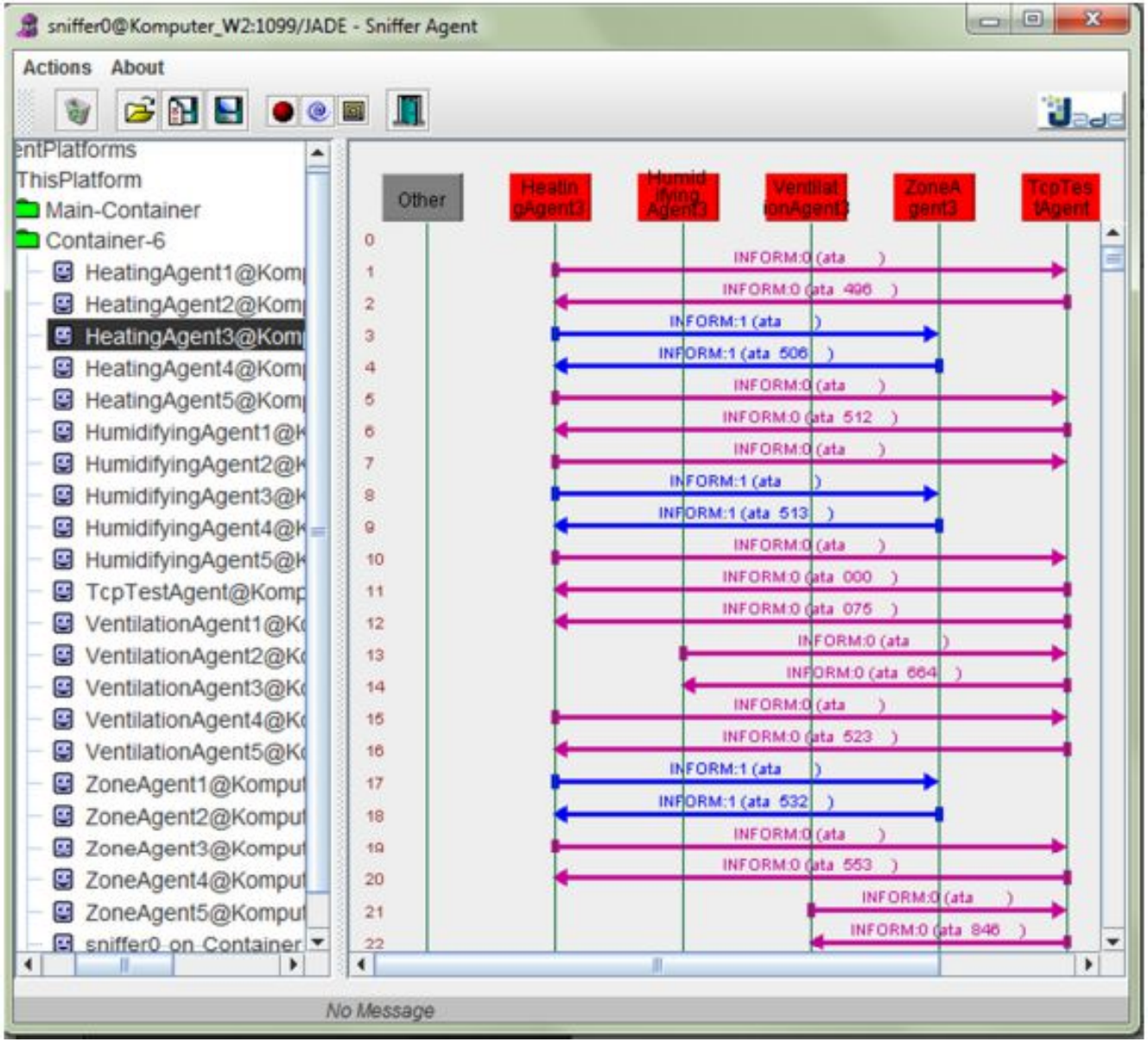
Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

```
<ZoneAgent1>: Received Request to turn the heating on
<HeatingAgent2>: Received temperature value from sensor agent: 11.742 °C
<HeatingAgent2>: Received message: Approved
<HeatingAgent2>: The setpoint temp is set to preferred 22.0 °C
<ZoneAgent2>: Received Request to turn the heating on
<HeatingAgent2>: Received temperature value from sensor agent: 11.843 °C
<HeatingAgent3>: Received temperature value from sensor agent: 11.984 °C
<ZoneAgent3>: Received Request to turn the heating on
<HeatingAgent3>: Received message: Approved
<HeatingAgent3>: The setpoint temp is set to preferred 20.0 °C
<HeatingAgent3>: Received temperature value from sensor agent: 11.984 °C
<HeatingAgent4>: Received temperature value from sensor agent: 12.791 °C
<HeatingAgent4>: Received message: Approved
<HeatingAgent4>: The setpoint temp is set to preferred 24.0 °C
<HeatingAgent4>: Received temperature value from sensor agent: 12.791 °C
<ZoneAgent4>: Received Request to turn the heating on
<HeatingAgent5>: Received temperature value from sensor agent: 11.821 °C
<ZoneAgent5>: Received Request to turn the heating on
<HeatingAgent5>: Received message: Approved
<HeatingAgent5>: The setpoint temp is set to preferred 20.0 °C
<HeatingAgent5>: Received temperature value from sensor agent: 11.821 °C
<VentilationAgent1>: Received humidity value from sensor agent: 100.0%
```

Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym

```
<ZoneAgent1>: ReceivedRequest to turn on the Ventilating
<ZoneAgent1>: Relative humidity limit for Fan is: 50.0%
<VentilationAgent1>: Received humidity value from sensor agent: 100.0%
<VentilationAgent2>: Received humidity value from sensor agent: 100.0%
<ZoneAgent2>: ReceivedRequest to turn on the Ventilating
<ZoneAgent2>: Relative humidity limit for Fan is: 50.0%
<VentilationAgent2>: Received humidity value from sensor agent: 100.0%
<VentilationAgent3>: Received humidity value from sensor agent: 100.0%
<ZoneAgent3>: ReceivedRequest to turn on the Ventilating
<ZoneAgent3>: Relative humidity limit for Fan is: 50.0%
<VentilationAgent3>: Received humidity value from sensor agent: 100.0%
<VentilationAgent4>: Received humidity value from sensor agent: 93.995%
<ZoneAgent4>: ReceivedRequest to turn on the Ventilating
<VentilationAgent4>: Received humidity value from sensor agent: 93.995%
<ZoneAgent4>: Relative humidity limit for Fan is: 50.0%
<VentilationAgent5>: Received humidity value from sensor agent: 100.0%
<ZoneAgent5>: ReceivedRequest to turn on the Ventilating
<ZoneAgent5>: Relative humidity limit for Fan is: 50.0%
<VentilationAgent5>: Received humidity value from sensor agent: 100.0%
```

Wieloagentowy system sterowania komfortem termicznym w budynku jednorodzinnym



Klasyczny Algorytm Genetyczny

Inspiracja dla algorytmów genetycznych/ewolucyjnych

- **Przekazywanie** przez organizmy pewnych, charakterystycznych dla nich **cech** (nabytych w wyniku wielopokoleniowego życia w pewnym środowisku) zapisanych w genach **kolejnym pokoleniom** w wyniku **reprodukcji**

- Cechy te zapisane są w **genach**, które przechowywane są w **chromosomach**, które z kolei tworzą strukturę **genotypu**
- **Fenotyp** to zestaw wartości odpowiadający danemu genotypowi (zdekodowany genotyp)

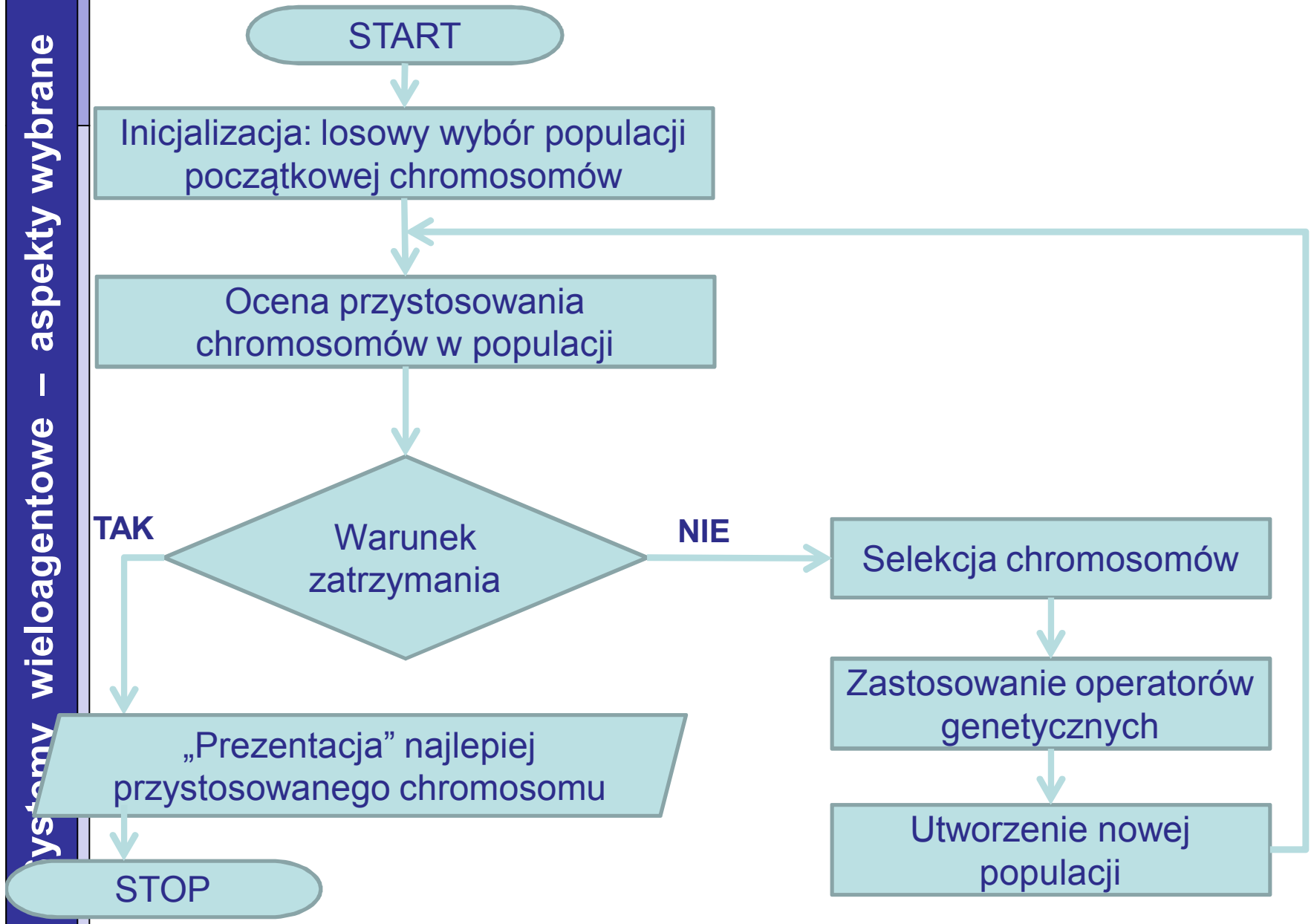
- Podczas przekazywania cech dochodzi do modyfikacji genów:
  - następuje **krzyżowanie** różniących się chromosomów obojga rodziców,
  - często zachodzi **mutacja**, czyli zamiana pojedynczych genów w chromosomie

Inspiracja dla algorytmów genetycznych/ewolucyjnych

- **Nowy organizm** ma cechy **swoich rodziców** ale i takie **charakterystyczne dla siebie**, które **decydują** o tym czy jest dobrze czy źle przystosowany do życia w danym środowisku
- Jeśli dobrze (**korzystna** kombinacja genów), **to przekaże** zapewne swój materiał genetyczny potomstwu
- Jeśli źle (niekorzystna kombinacja genów), **to trudno będzie mu żyć** w danym środowisku i **trudno mu będzie przekazać** swoje geny potomstwu

Ale jak to wykorzystać w obliczeniach numerycznych?

Klasyczny algorytm genetyczny



Rój, Stado, Ławica, Kolonia…

Wprowadzenie ……..☺

https://www.youtube.com/watch?v=DmO4Ellgmd0

https://www.youtube.com/watch?v=dqiOrobSVFo

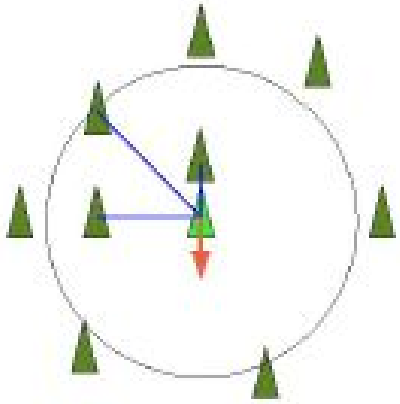
Rój, Stado, Ławica, Kolonia…

Rój, Stado, Ława, Kolonia…

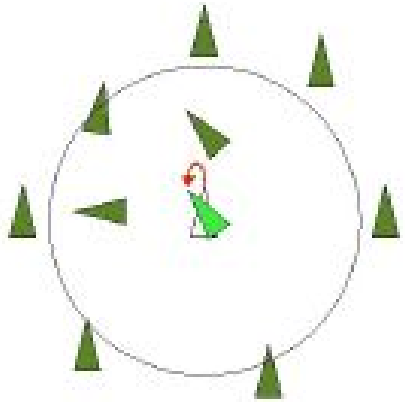
Rój, Stado, Ławica, Kolonia…

Rój, Stado, Ławica, Kolonia…

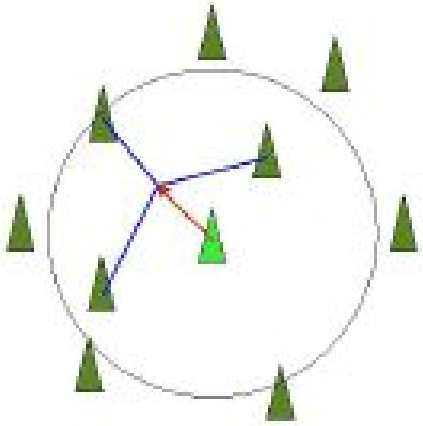
- Podstawowy model zachowania osobnika w grupie



Separacja:
Gdy za blisko to oddal się od sąsiada



Wyrównanie kierunku:
W sensie średnim do swoich sąsiadów



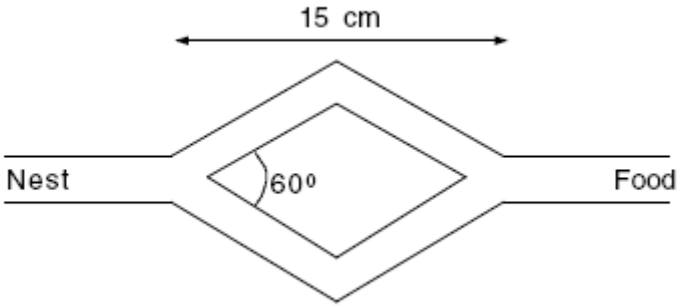
Wyrównanie położenia:
W sensie średnim do swoich sąsiadów

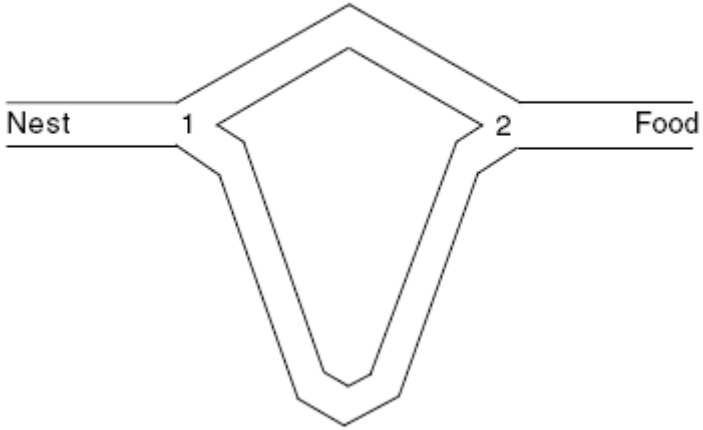
Rój, Stado, Ławica, Kolonia…

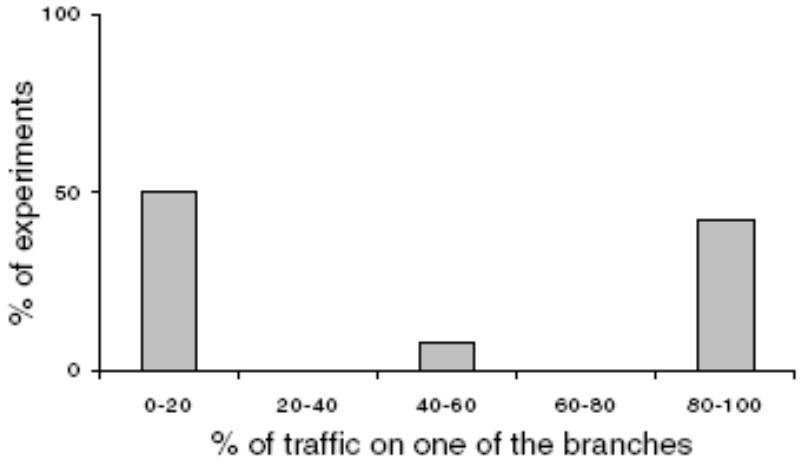
Rój, Stado, Ławica, Kolonia…

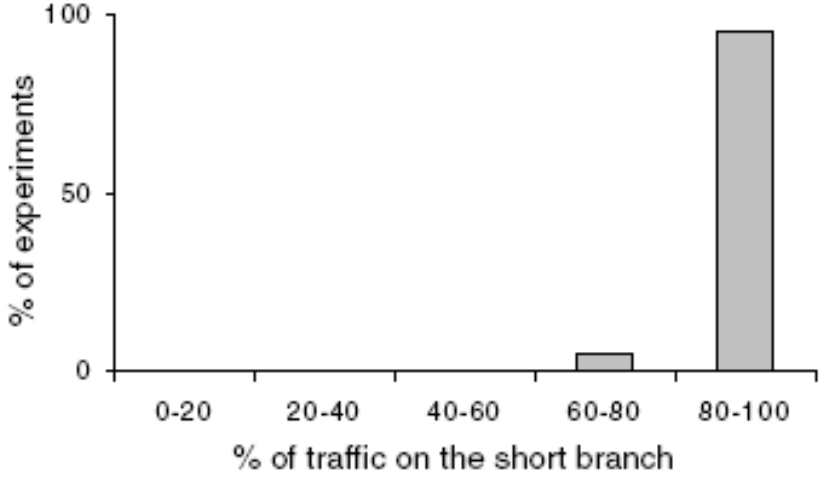
- A jak postępują mrówki? -> feromony

Eksperyment na prawdziwych mrówkach





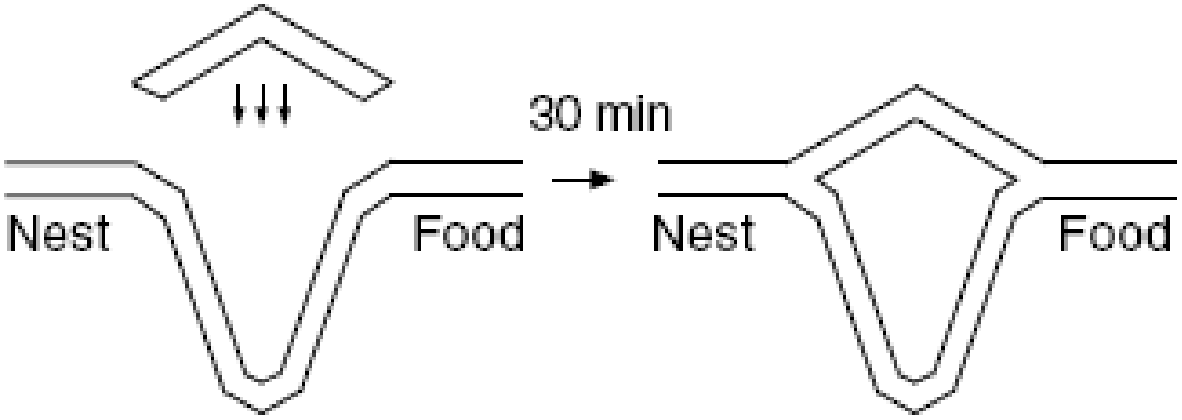


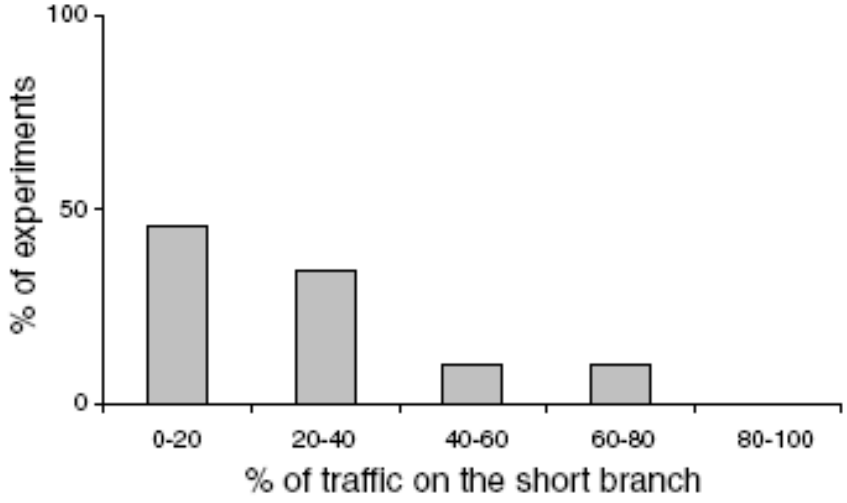


Rój, Stado, Ławica, Kolonia…

- A jak postępują mrówki? -> feromony

Eksperyment na prawdziwych mrówkach





Ogólny model cząstki (osobnika) w grupie (stadzie)

Każda cząstka (osobnik):

- posiada określone położenie (współrzędne),
- zna swoje położenie,
- zna wartość funkcji ewaluacyjnej (funkcji celu) dla swojego położenia,
- ma ustaloną prędkość i zwrot,
- pamięta najlepsze położenie jakie dotychczas osiągnęła,
- pamięta wartość funkcji ewaluacyjnej (funkcji celu) dla swojego najlepszego położenia,
- zna swoich sąsiadów,
- zna najlepsze uzyskane położenia swoich sąsiadów,
- zna wartość funkcji ewaluacyjnej (funkcji celu) dla najlepszych położeń swoich sąsiadów.

Ten zestaw cech daje umiejętność skupiania się osobników czy zdolność znajdowania określonych punktów w przestrzeni.

Rój Cząstek

Rój Cząstek – inspiracje

- Dynamika poruszającego się stada, którą można wykorzystać do przeszukania przestrzeni rozwiązań rozważanego zadania

Rój Cząstek – podstawowe kroki algorytmu

1. Inicjalizacja położenia, obliczenie wartości funkcji ewaluacyjnych, wyznaczenie najlepszego (globalnie) znalezionego rozwiązania w domenie reprezentującej obszar rozwiązań analizowanego zadania *(prędkość i kierunek cząstek przyjmuje się na tym etapie jako zerowy)*
2. Wyznaczenie nowych prędkości i kierunków przemieszczania się cząstek roju,
3. Przemieszczenie cząstek w nowe położenie,
4. Obliczenie wartości funkcji ewaluacyjnych dla nowych współrzędnych cząstek,
5. Sprawdzenie, czy nowe położenie cząstek jest lepsze od najlepszych dotychczas znanych położeń,
6. Zapamiętanie nowych, najlepszych położeń cząstek,
7. Wyznaczenie najlepszego (globalnie) znalezionego rozwiązania po zakończonej pojedynczej iteracji.
8. Sprawdzenie warunku stopu, jeżeli nie spełniony powrót do drugiego kroku algorytmu,
9. Prezentacja najlepszego rozwiązania i zakończenie algorytmu.

Rój Cząstek – ustalenie kierunku ruchu i prędkości cząstek

Cząstka może przemieścić się w:

- dotychczas obranym kierunku,
- kierunku swojego najlepszego położenia,
- w kierunku najlepszego znanego położenia sąsiadów.

$$v^{t+1} = c_1 r_1 v^t + c_2 r_2 \left(y^t - x^t\right) + c_3 r_3 \left(y^{*t} - x^t\right)$$

v^t – prędkość cząstki w chwili t,

x^t – położenie cząstki w chwili t,

y^t – najlepsza znana pozycja cząstki w chwili t,

y^{*t} – najlepsza znana pozycja sąsiadów cząstki w chwili t,

c_1 – współczynnik określający jak bardzo cząstka ufa swojemu dotychczasowemu kierunkowi,

c_2 – współczynnik określający jak bardzo cząstka ufa kierunkowi do swojego najlepszego położenia,

c_3 – współczynnik określający jak bardzo cząstka ufa kierunkowi do najlepszego znanego położenia sąsiadów,

r_1 – losowy współczynnik decydujący o prędkości poruszania się w kierunku wyznaczonym przez c_1,

r_2 – losowy współczynnik decydujący o prędkości poruszania się w kierunku wyznaczonym przez c_2,

r_3 – losowy współczynnik decydujący o prędkości poruszania się w kierunku wyznaczonym przez c_3.

Rój Cząstek – przemieszczanie się cząstek w przestrzeni

Przemieszczanie cząstek następuje zgodnie z zależnością:

$$x^{t+1} = x^t + v^{t+1}$$

Rój Cząstek

Przykłady zastosowań:

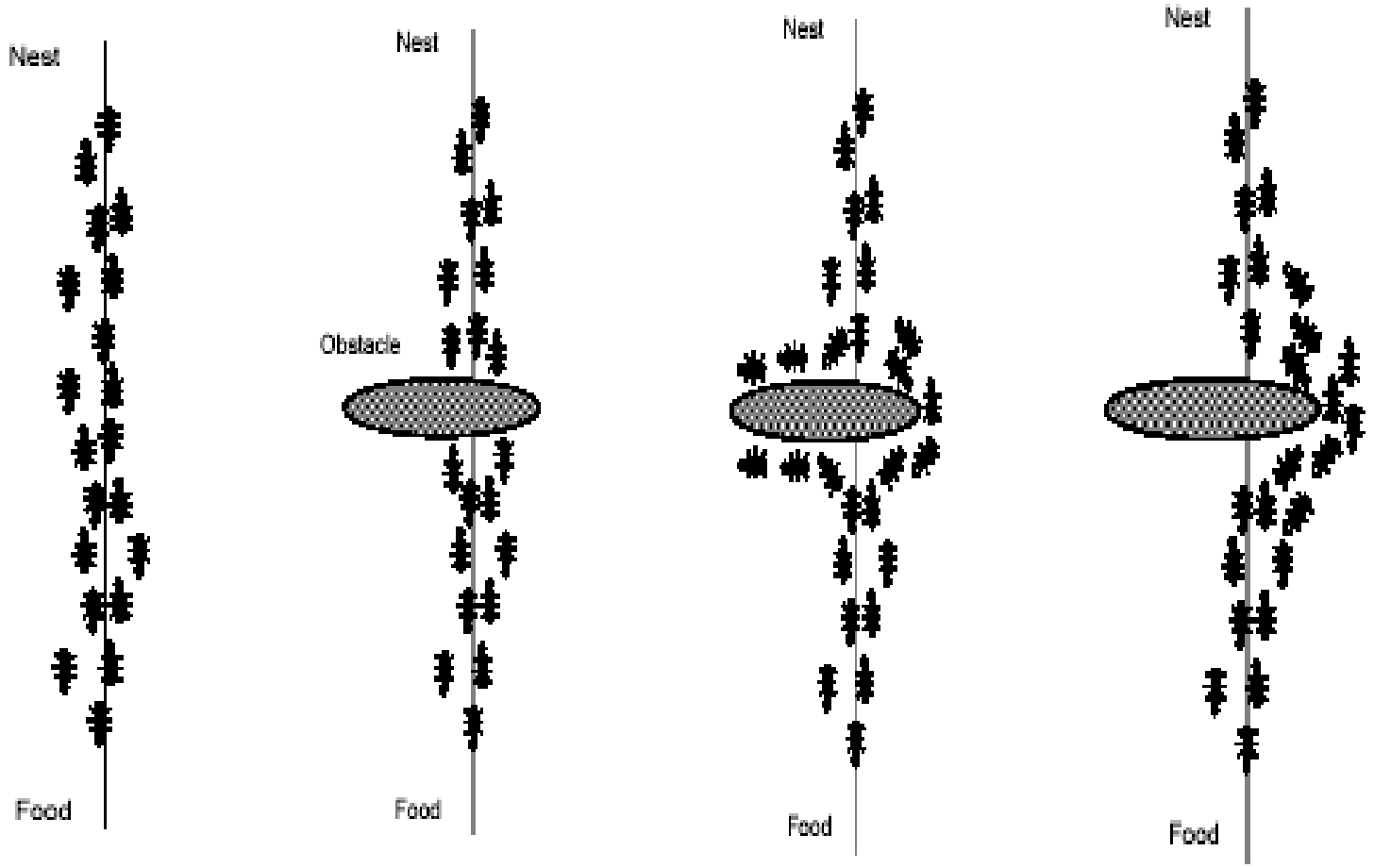
- ….

DEMO 1

DEMO 2

Kolonia Mrówek

Kolonia Mrówek - inspiracja



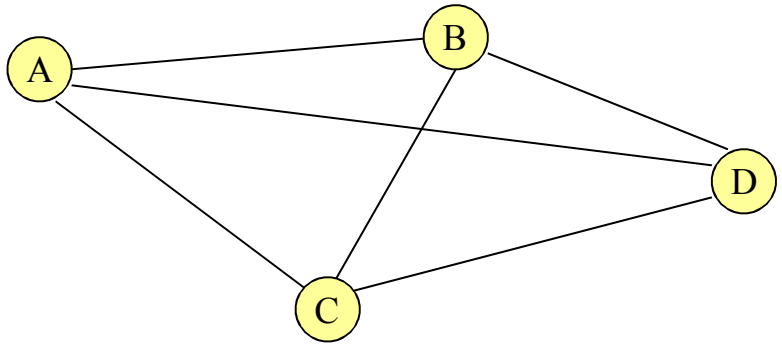
Kolonia Mrówek - podstawowe kroki algorytmu

1. Generowanie rozwiązania w przestrzeni rozwiązań -> rozchodzenie się mrówek wokół mrowiska
2. Aktualizacja ścieżek feromonów
3. Sprawdzenie warunku stopu, jeżeli nie spełniony powrót do pierwszego kroku algorytmu (mrówki ruszają od początku)
4. Prezentacja najlepszego rozwiązania i zakończenie algorytmu

Kolonia Mrówek - podstawowe kroki algorytmu

1. Generowanie rozwiązania w przestrzeni rozwiązań -> rozchodzenie się mrówek wokół mrowiska

typowo przeszukiwanie struktury grafu



Kolonia Mrówek - podstawowe kroki algorytmu

A

B

C

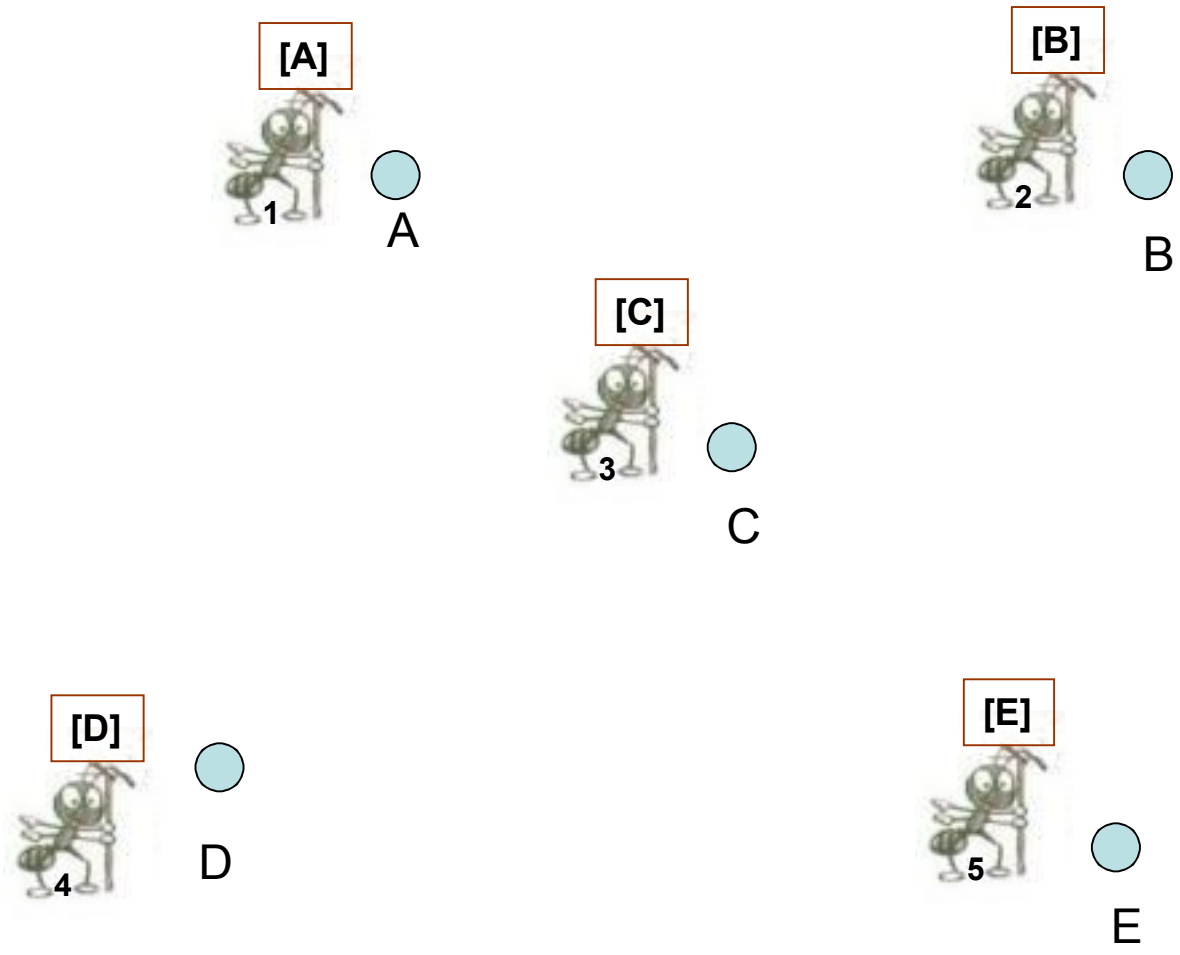
D

E

d_{AB} =100;d_{BC} = 60…;d_{DE} =150

Iteracja 1

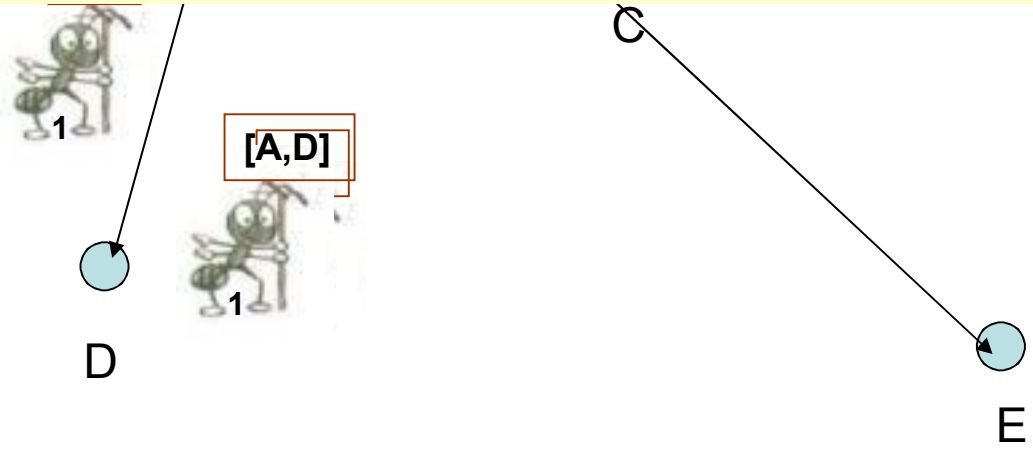
Kolonia Mrówek - podstawowe kroki algorytmu



Jaką drogę wybrać?

Kolonia Mrówek - podstawowe kroki algorytmu

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\sigma_{ij}]^{\beta}}{\sum_{h \notin tabu_k} [\tau_{ih}(t)]^{\alpha}[\sigma_{ih}]^{\beta}} & \text{jeśś } j \in tabu_k \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$



Kolonia Mrówek - podstawowe kroki algorytmu

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\sigma_{ij}]^{\beta}}{\sum_{h \notin tabu_k} [\tau_{ih}(t)]^{\alpha}[\sigma_{ih}]^{\beta}} & \text{jeśś j} \in \text{tabu}_{\text{k}} \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$

τ_{ij} to ilość feromonu na łuku ze stanu i do stanu j

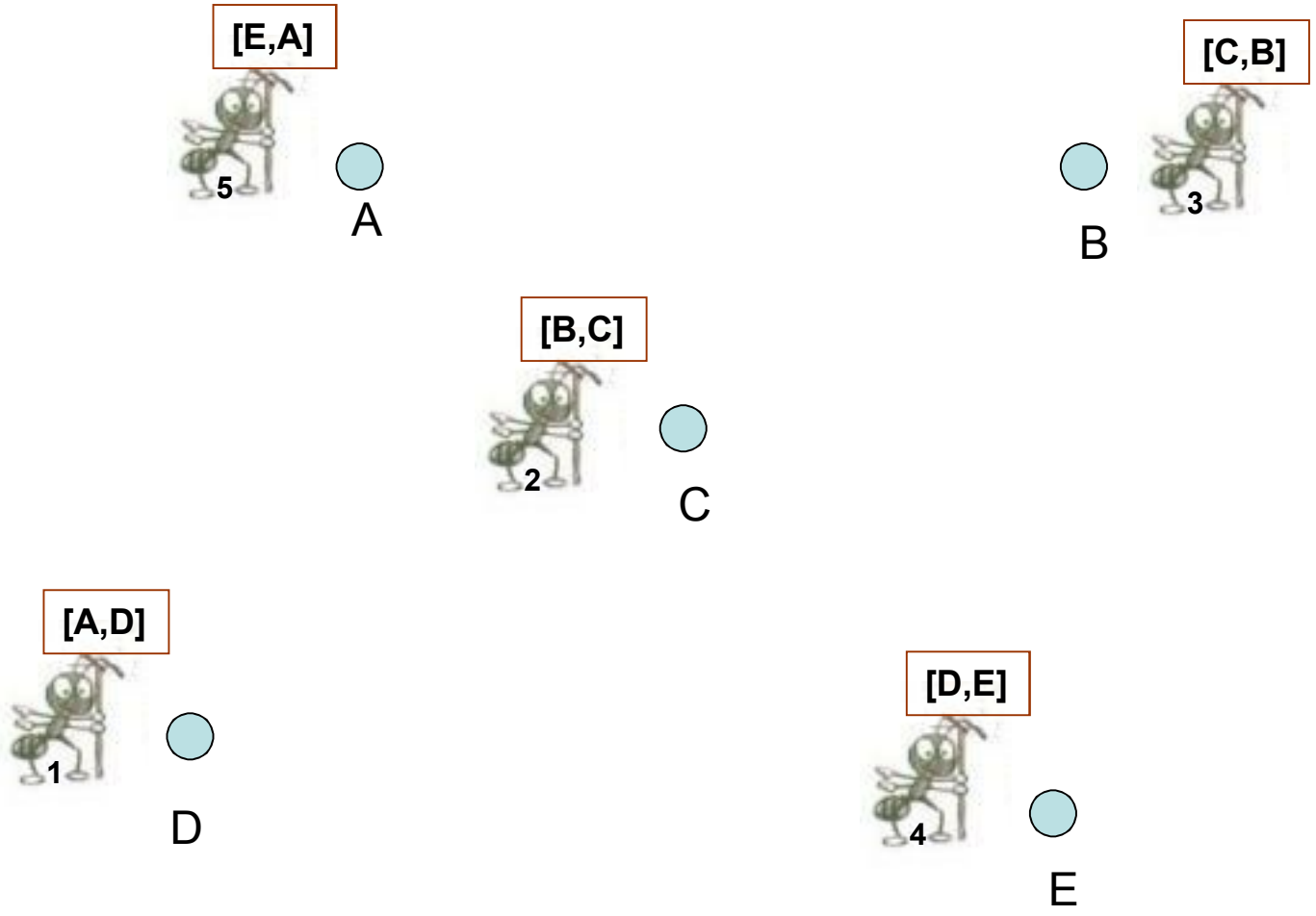
σ_{ij} to funkcja wskazująca tak zwaną „atrakcyjność" przejścia z i do j

α, β to współczynniki istotności powyższych dwóch wpływów na decyzję mrówki

$tabu_k$ to zbiór odwiedzonych już przez mrówkę węzłów w danej iteracji

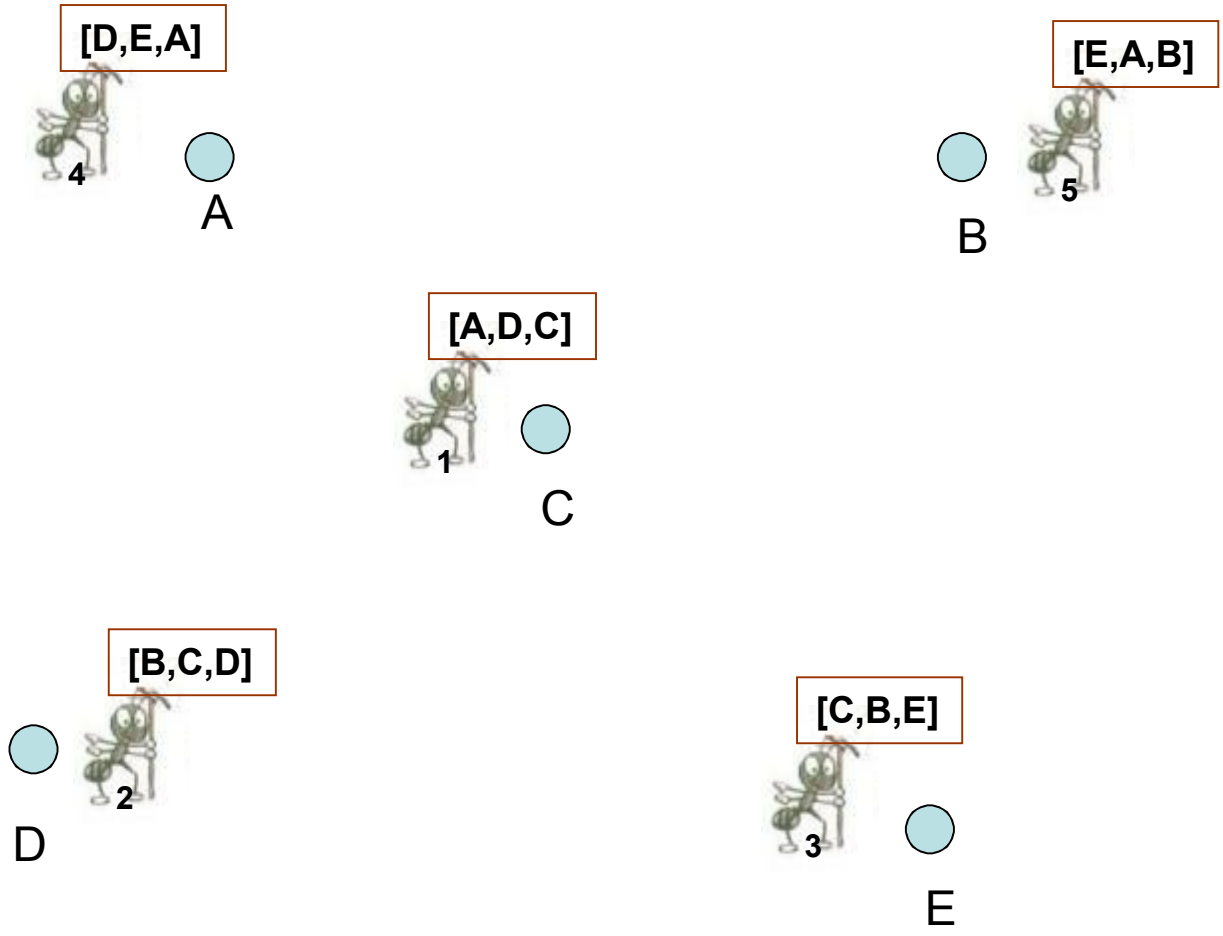
Iteracja 2

Kolonia Mrówek - podstawowe kroki algorytmu



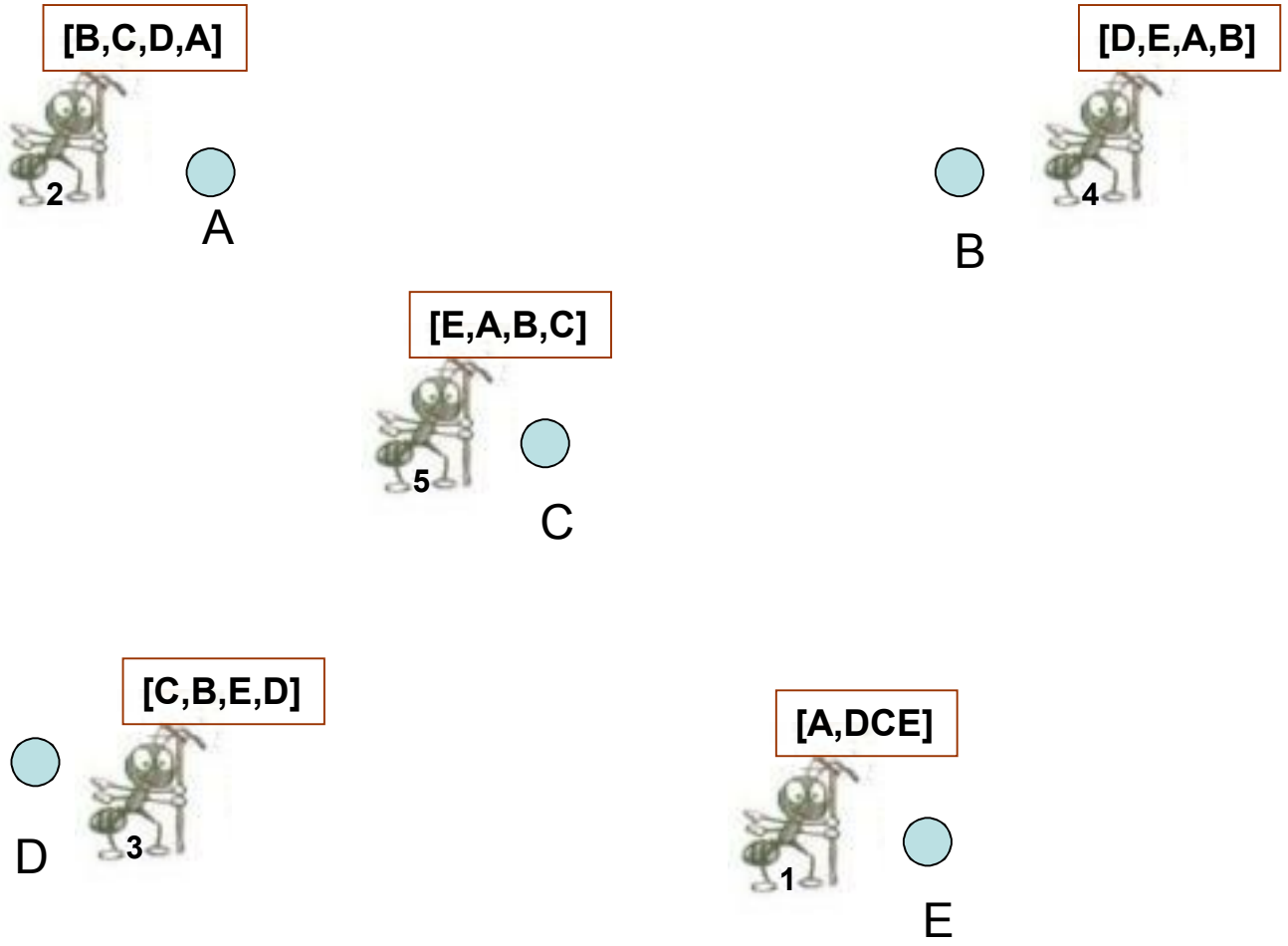
Iteracja 3

Kolonia Mrówek - podstawowe kroki algorytmu



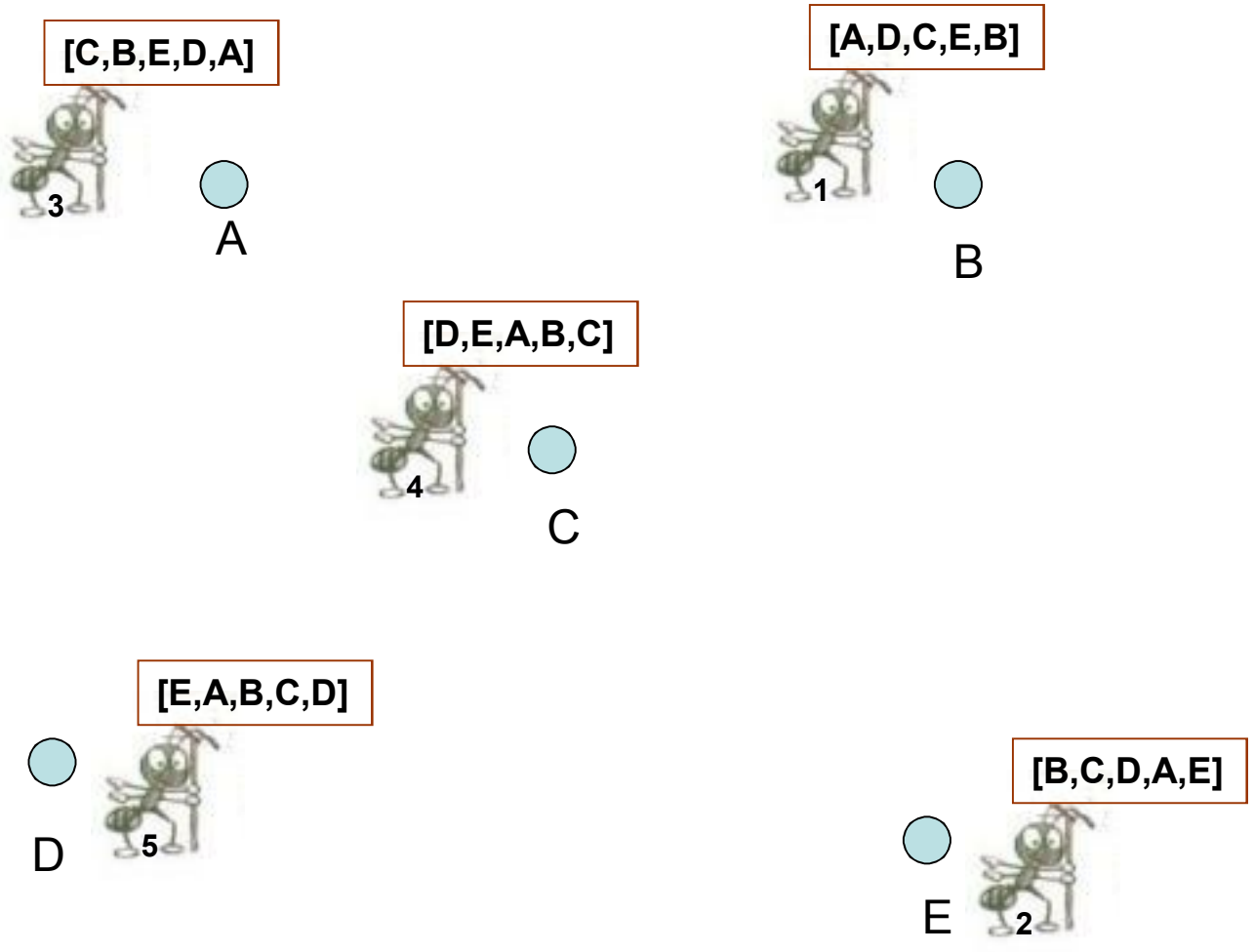
Iteracja 4

Kolonia Mrówek - podstawowe kroki algorytmu



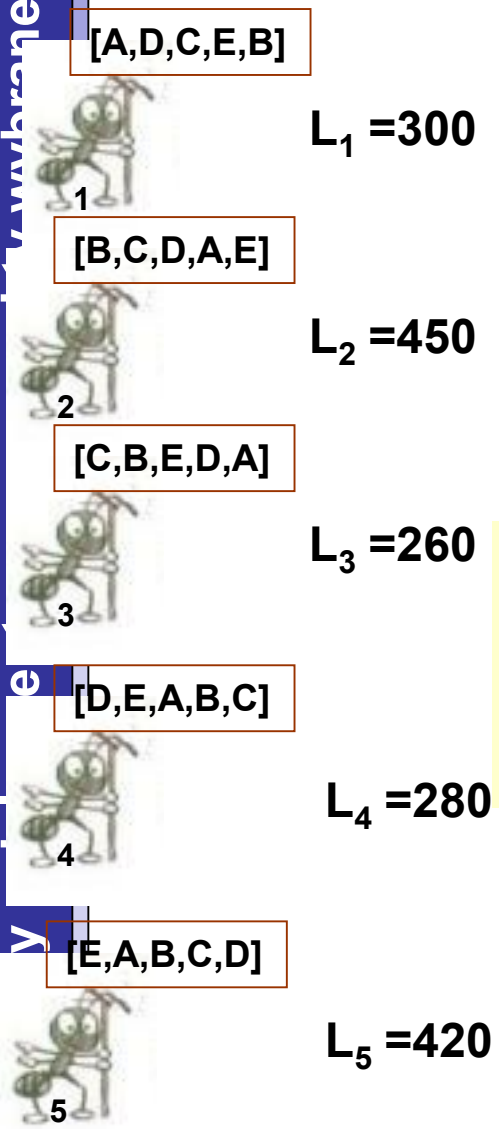
Iteracja 5

Kolonia Mrówek - podstawowe kroki algorytmu



Aktualizacja ścieżek feromonów

Kolonia Mrówek - podstawowe kroki algorytmu



$$\tau_{i,j}^{k} = (1-\rho)\cdot\tau_{i,j}^{k} + \Delta\tau_{i,j}^{k}$$

ρ to współczynnik wyparowywania feromonu

$$\Delta\tau_{i,j}^{k} = \begin{cases} \dfrac{Q}{L_k} \; if\,(i,j) \in \text{naljlepsz ścieżka mrówki k} \\ 0 \qquad\qquad \text{w przeciwnym przypadku} \end{cases}$$

Q to pewna dobrana stała (parametr procedury)

L_k to koszt rozwiązania związanego z mrówką k

Powrót mrówek do pozycji początkowych, może pójdą inną drogą

Kolonia Mrówek

Przykłady zastosowań:

- ….

DEMO KM1

DEMO KM2

Bibliografia

- Wooldridge, Michael . An Introduction to MultiAgent Systems. John Wiley & Sons, 2002.
- Shoham, Yoav; Leyton-Brown, Kevin. Cambridge University Press, 2008.
- J. H. Holland. Adaptation in Natural and Artificial Systems. The University of Michigan Press, Ann Arbor, Michigan, 1975.
- Z. Michalewicz. Genetic Algorithms + Data Structures = Evolution Programs. *Springer-Verlag, Berlin, third edition, 1996.*
- James Kennedy, Russell Eberhart. Particle swarm optimization. In Proceedings of IEEE International Conference on Neural Networks. IEEE Press (strony 1942–1948), NJ, USA, 1995.
- *Marco Dorigo, Thomas Stützle. Ant Colony Optimization. The MIT Press, London, 2004.*

Dziękuję za uwagę !!!